9

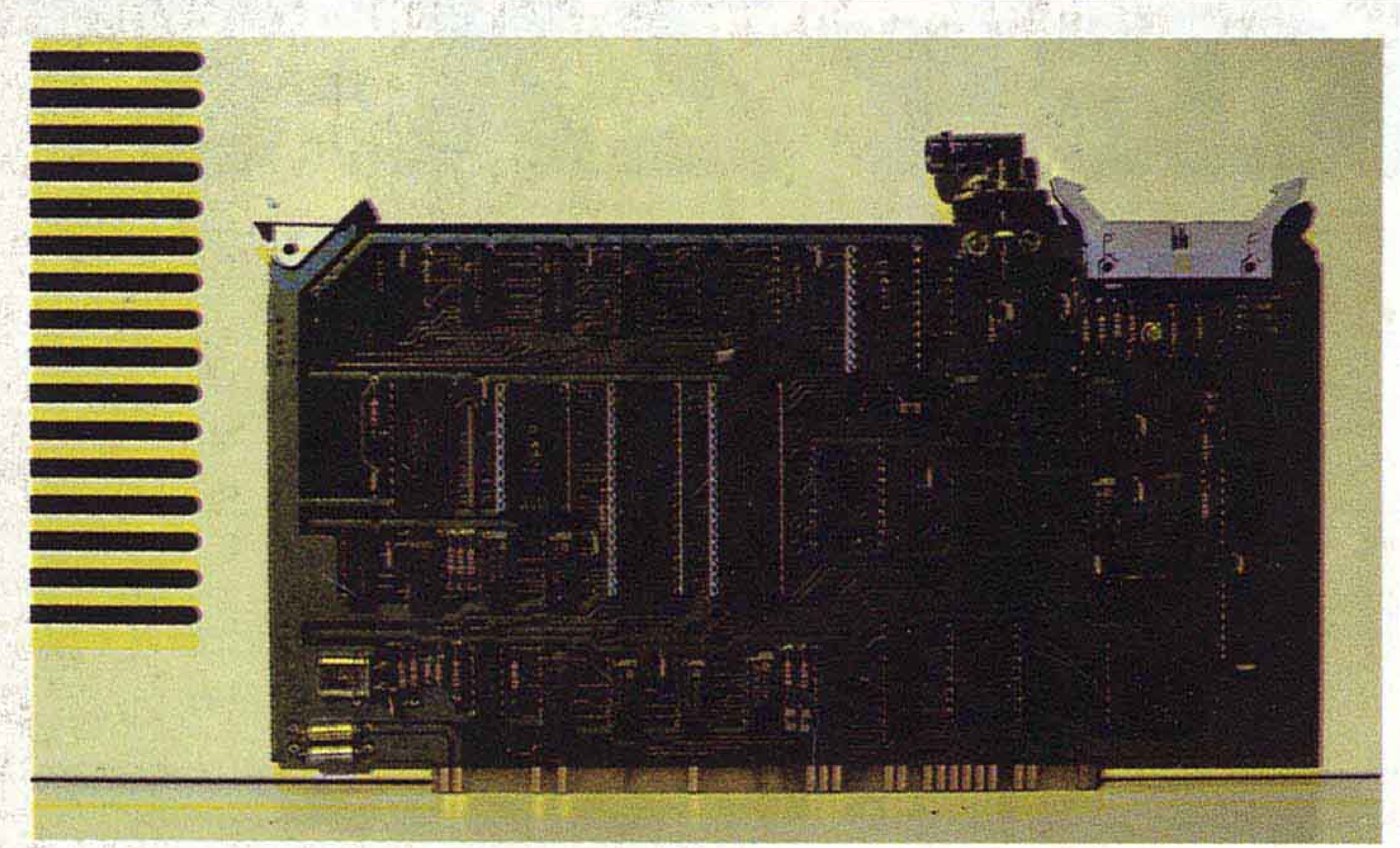
Enciclopedia Práctica de la

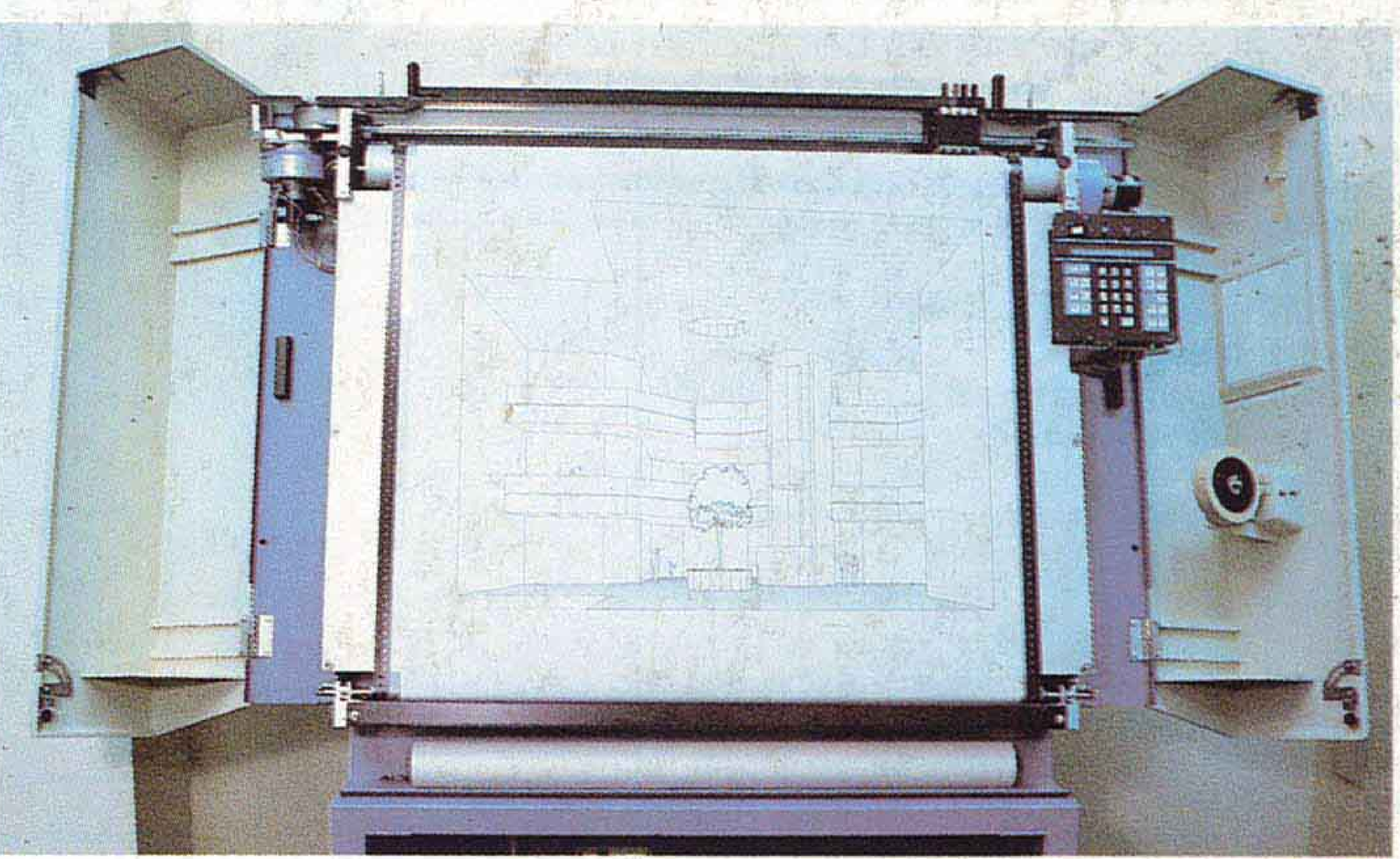
INFORMATICA

UNIDADES DE ENTRADA/SALIDA / HARDWARE: OSBORNE 1

LOS LENGUAJES MAQUINA / PLOTTERS

VISICALC / EL TELETRABAJO





Nueva Lente / Ingelek

125 ptas.

UNIDADES DE ENTRADA/SALIDA

LA función de las unidades de entrada/salida —E/S o I/O, según utilicemos notación castellana o anglosajona INPUT/OUTPUT— es adaptar la información procedente del exterior para que sea interpretable por el ordenador, así como adaptar la información suministrada por el ordenador para que pueda ser tratada por los periféricos.

Intercambio de información con el exterior

Las unidades periféricas no forman parte de la unidad central de proceso, de ahí que, necesariamente, haya que habilitar el intercambio de información entre la CPU y los periféricos.

Si comparamos al ordenador con el cuerpo humano, la CPU es el cerebro, mientras que las unidades periféricas más importantes serían los órganos en los que residen los cinco sentidos. De poco serviría tener un cerebro privilegiado, si la información obtenida a través de cualquier sentido (la vista, por ejemplo), no fuera interpretable por aquél.

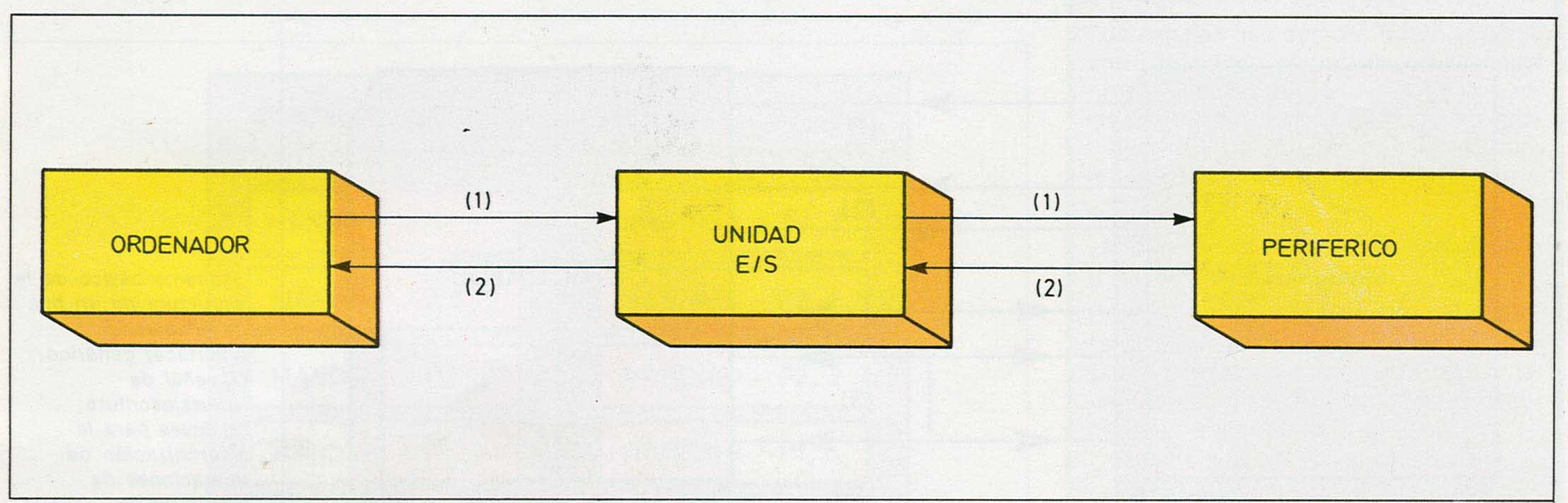
Este papel está reservado al sistema nervioso en el cuerpo humano y a las unidades de E/S en el ordenador.

Justificación de las unidades de entrada/salida

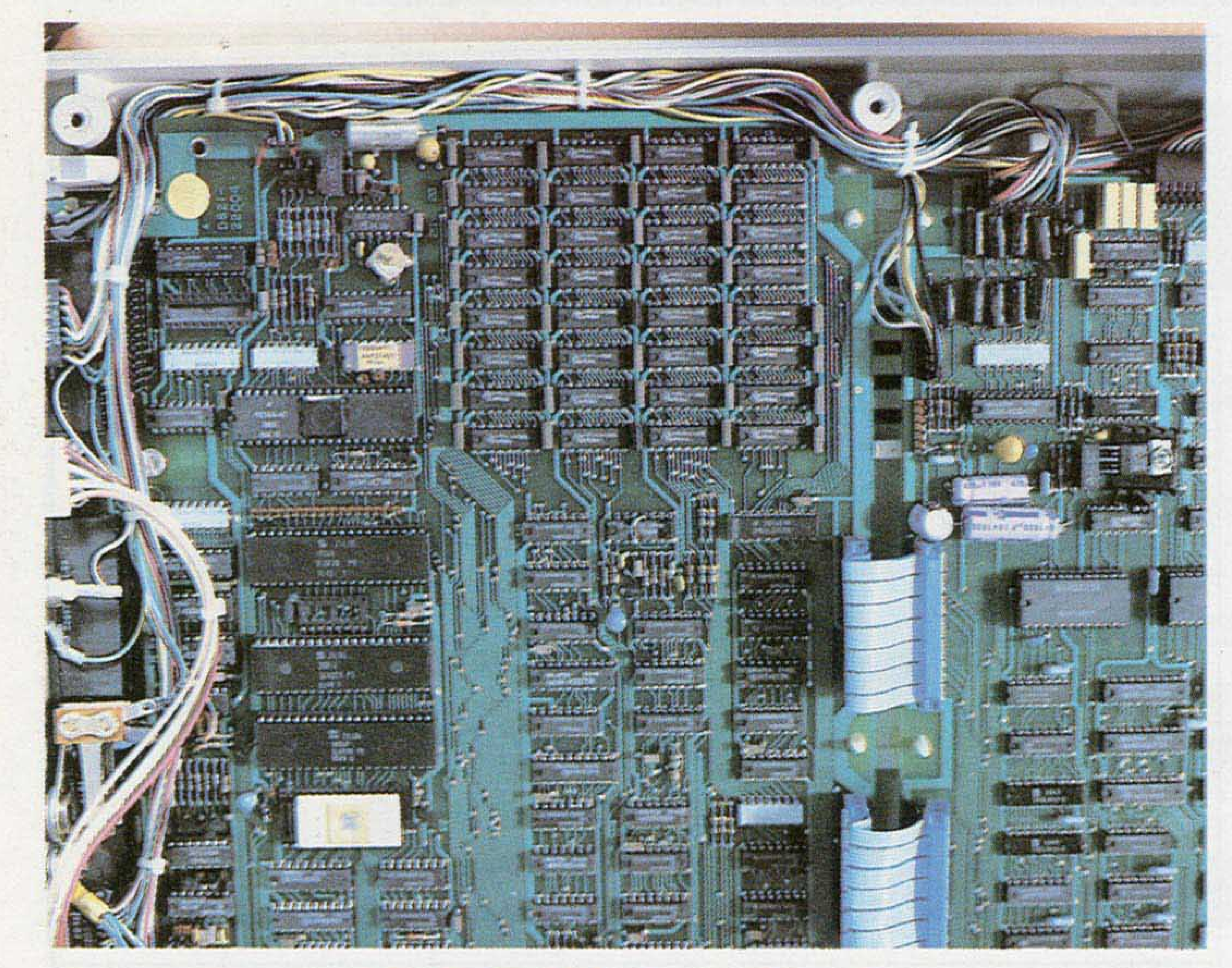
Estas unidades sirven para canalizar las transferencias de información entre el ordenador, mini o micro, y los dispositivos periféricos exteriores. Por supuesto, su actuación es controlada por la CPU.

Un ordenador puede disponer de varias unidades de E/S que, a su vez, pueden controlar varios periféricos del mismo tipo. Las principales ventajas obtenidas con su empleo son las siguientes:

1. La velocidad de trabajo de la CPU es muy superior a la de los periféricos.



Las unidades de entrada/salida se ocupan de transformar la información representada en formato del ordenador a información interpretable por el dispositivo periférico (1) y viceversa (2).



En los microordenadores actuales, las unidades de entrada/salida suelen estar constituidas por circuitos integrados programables especializados en este tipo de tareas.

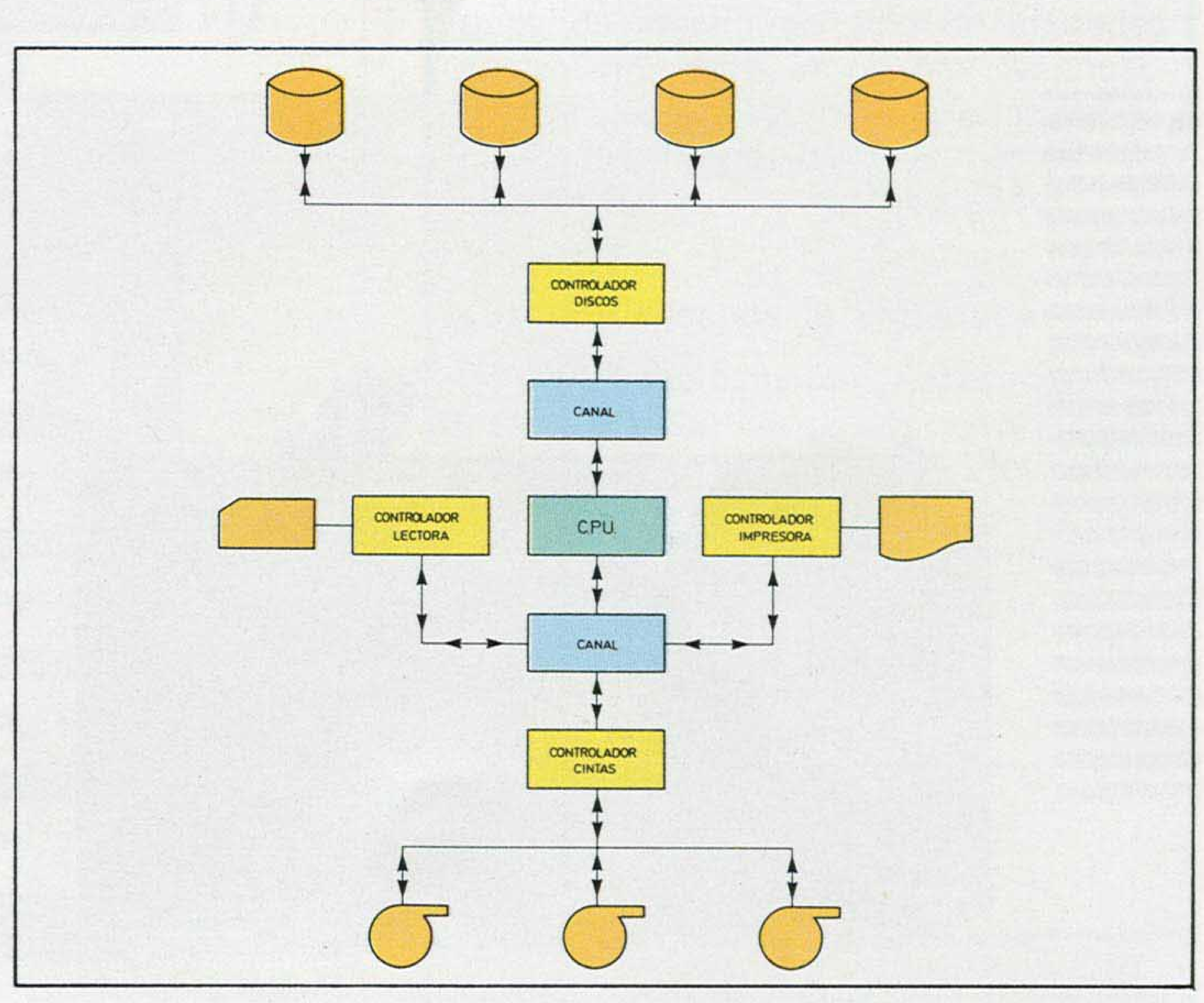


Diagrama de la arquitectura de la unidad central de un ordenador en el que intervienen los dos tipos genéricos de unidades de E/S: canales y controladores de periféricos.

UNIDADES DE ENTRADA/SALIDA

Mediante las unidades de E/S se consigue la independencia entre ambas.

2. Las unidades periféricas pueden tener distinta forma de tratar la información, incluso dentro de unidades del mismo tipo esta característica varía según los fabricantes. Mediante las unidades de E/S se pueden adaptar muy diversos tipos de periféricos, independientemente de que sus formatos sean distintos a los del ordenador.

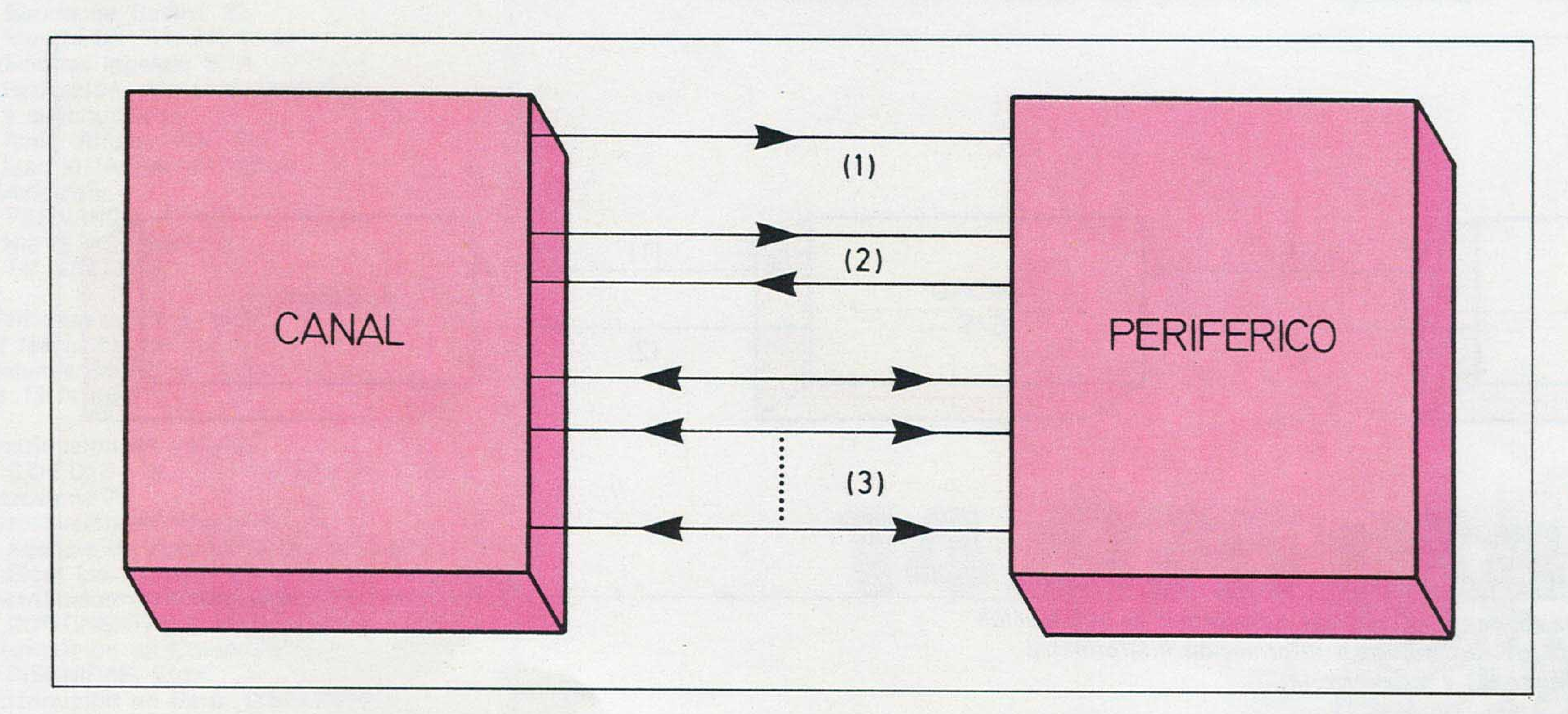
3. Las unidades de E/S también sirven de intermediarias entre las lógicas binarias del ordenador y de los periféricos, que pueden ser distintas.

Tipos de unidades de E/S

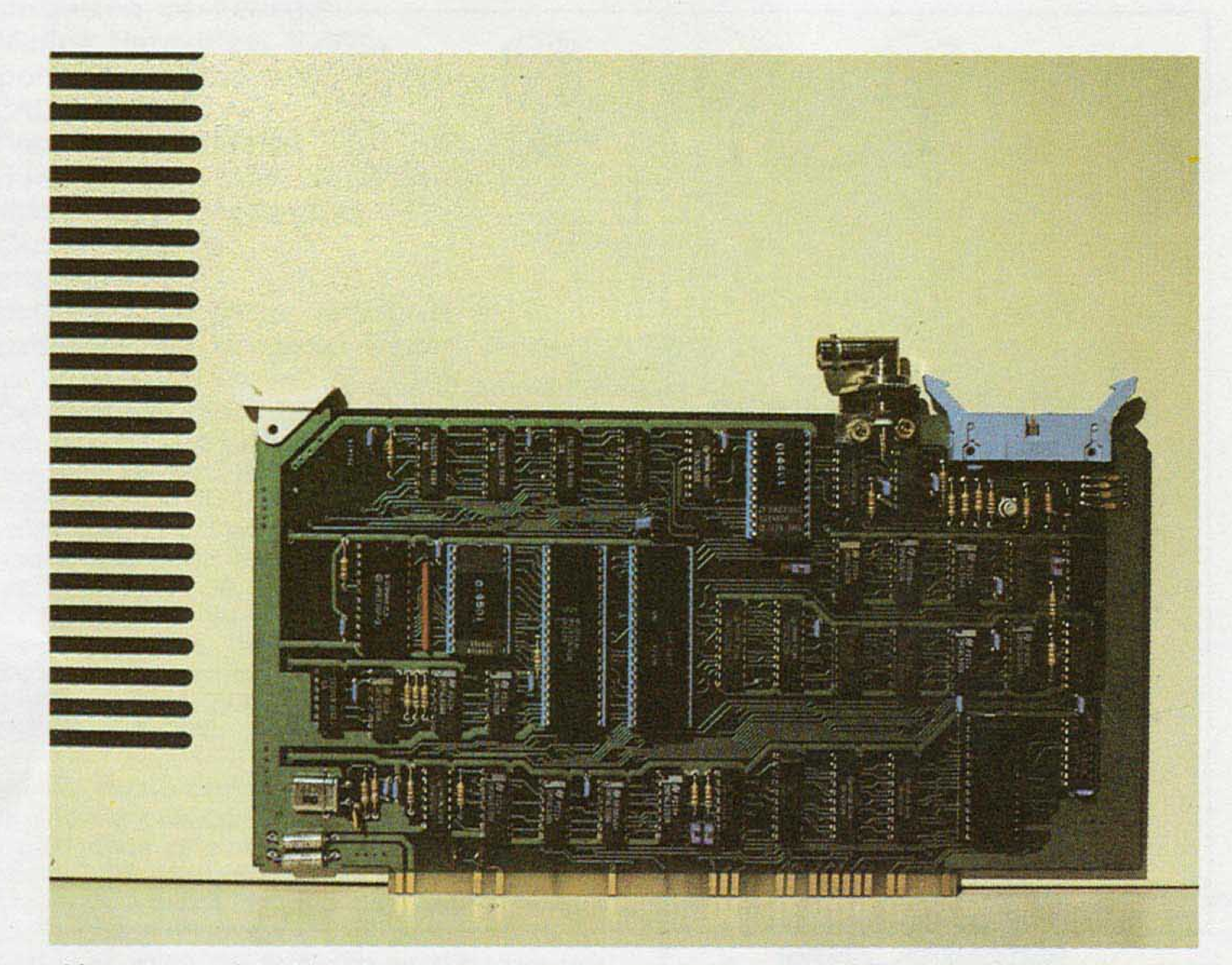
El intercambio de información entre la memoria del ordenador y el exterior se realiza, normalmente, a través de dos tipos de unidades de E/S: los canales y los controladores de periféricos.

Canales

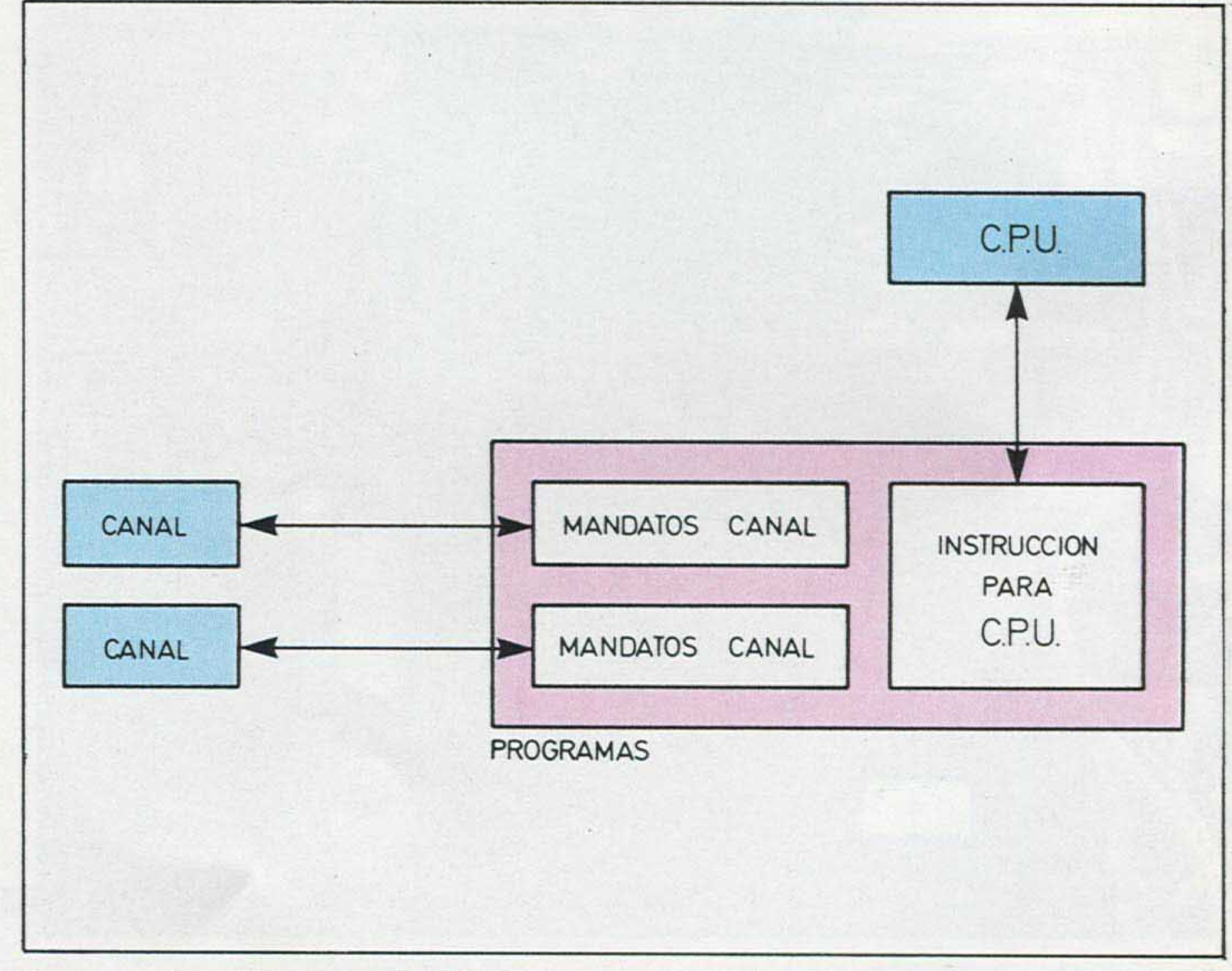
El número de operaciones por segundo que puede ejecutar cualquier ordenador es muy superior al de transferencias de información por segundo, esto obligaba a mantener bloqueada a la CPU mientras se realizaban las operaciones de entrada/salida. La solución a este problema fueron los *canales*. Cuando la CPU necesita realizar una transferencia de información con un periférico lento, por ejemplo, una impresora, no tiene por qué esperar a que ésta termine de escribir una línea para ordenar la escritura de la siguiente. Simplemente, «lanza» todas las órdenes de escritura al canal y continúa ejecutando otras instrucciones del programa o incluso dándolo por acabado. Será responsabilidad del canal gestionar adecuadamente todas las operaciones de salida que le han sido transferidas.



Esquema básico de la estructura de un bus de adaptación (interface) genérico: (1) señal de lectura/escritura; (2) líneas para la sincronización de operaciones de transferencia; (3) líneas de transferencia de información.



Algunas unidades de entrada/salida pueden requerir una circuitería electrónica compleja. En la fotografía aparece una unidad de E/S encargada de la adaptación de un puesto de trabajo al sistema Secoinsa Serie-20.



Los canales liberan a la CPU de tareas supletorias; mientras la CPU ejecuta instrucciones, los canales pueden ocuparse de gestionar las operaciones de transferencia con los periféricos.

Este tipo de dispositivos sólo se utilizan en ordenadores y miniordenadores. La mayoría de los microordenadores no disponen de canales, realizándose el intercambio de información directamente entre la unidad de control de la CPU y los controladores de periféricos.

Controladores de periféricos

Se encargan de gestionar una o varias unidades periféricas de un mismo tipo; para ello tienen que ser capaces de:

- Interpretar las instrucciones que reciben o entregan del/al ordenador. Esto se realiza a través de circuitos que adaptan y reconocen las señales de «interface» del canal o de la unidad central de proceso.
- Controlar al periférico asociado según sus características; para ello decodifican la operación que se les ordena ejecutar (lectura, escritura, rebobinado...) y, en el sentido inverso, emiten información del estado del periférico (ocupado, preparado, rebobinado...).

La complejidad de los controladores de periféricos puede ser muy distinta y suele estar en consonancia con la complejidad del propio periférico. Evidentemente, es muy distinto controlar una lectora de tarjetas, que realiza necesariamente un trabajo secuencial y

(1)

ORDEN DE LLEGADA	ORDEN DE SALIDA
1 MANDATO – P1.1	MANDATO – P1.1 1
2 MANDATO – P2.1	MANDATO – P2.1 2
3 MANDATO – P2.2	MANDATO – P2.2 3
4 MANDATO – P1.2	MANDATO – P1.2 4
5 MANDATO – P2.3	MANDATO – P2.3 5

(2)

ORDEN DE LLEGADA	ORDEN DE SALIDA
1 MANDATO – P1.1	MANDATO – P2.1 1
2 MANDATO – P2.1	MANDATO – P2.2 2
3 MANDATO – P2.2	MANDATO – P2.3 3
4 MANDATO – P1.2	MANDATO – P1.1 4
5 MANDATO – P2.3	MANDATO – P1.2 5

Secuencia de órdenes de salida, para un mismo orden de llegada, según se aplique la alternativa de prioridad «primero en llegar, primero en salir» (1) o «prioridad exterior» (2), en este último caso, se considera prioritario al procesador P2.

Glosario

¿Cuáles son las principales ventajas introducidas por las unidades de E/S?

1. Desbloqueo de la CPU al encargarse de forma autónoma de gestionar las transacciones de información.
2. Adaptación de los distintos formatos con que trabajan los periféricos a los formatos del ordenador.
3. Control y adaptación de las distintas lógicas binarias de los periféricos y la CPU.

¿Qué tipos de unidades de E/S existen?

Los *canales* que son los intermediarios entre la unidad central de proceso del ordenador y los periféricos, y los *controladores de periféricos* que pueden gestionar la utilización de uno o varios periféricos del mismo tipo.

¿Los controladores de periféricos son parte de los periféricos?

En algunos casos, cuando el controlador se encarga de un único periférico, se le puede considerar como parte del periférico, aunque, en general, el controlador es independiente del periférico.

¿Qué misiones específicas tienen encomendadas los controladores de periféricos?

Depende mucho del tipo de unidades controladas, pero, en general, se encargan de ejecutar las instrucciones recibidas de la CPU y de devolver a ésta información del estado del periférico.

¿En qué consiste la «Interface»?

Son las especificaciones necesarias para poder conectar los periféricos a las unidades de control o canales. Por extensión, suele denominarse de esta forma al dispositivo o unidad que se ocupa de adaptar y establecer este tipo de comunicaciones.

UNIDADES DE ENTRADA/SALIDA

exclusivamente de entrada de datos, que controlar una unidad de discos que puede ser utilizada tanto para entrada como para salida de datos e, incluso, de forma no secuencial. En los casos más complejos se puede llegar a utilizar microprocesadores para realizar la labor de controlar los periféricos.

«Interface» entre la CPU y las unidades periféricas

Se denomina «Interface» a las especificaciones de conexión necesarias para adaptar las unidades periféricas a la CPU o a los canales. En un sentido más amplio, también se entiende por «Interface» a todos los componentes necesarios para adaptar las señales del ordenador a las de los periféricos y viceversa. La «Interface» está compuesta fundamentalmente por dos tipos de hilos: los que llevan la información a transcribir y los que se encargan de sincronizar las operaciones e indicar si éstas son de entrada o salida.

Prioridades de acceso

En el caso de que dos o más procesadores soliciten al mismo tiempo un periférico, la unidad de E/S tiene varias alternativas para decidir a cuál de ellos atenderá primero:

1. *Primero en llegar, primero en salir*
Este método es el más sencillo, consiste en asignar los recursos según el orden en que han sido solicitados. La unidad de E/S mantiene una cola de transacciones a efectuar en el orden de llegada y las va realizando en el mismo orden.
2. *Prioridad exterior*
El sistema acepta unas prioridades marcadas desde el exterior, de forma que atenderá las solicitudes antes o después según tengan más o menos prioridad, respectivamente. El defecto de este procedimiento es que un procesador con máxima prioridad, que tenga muchas transacciones, puede llegar a bloquear el sistema.
3. *Asignación cíclica*
Consistente en atender, cíclicamente y siempre en el mismo orden, a todos los procesadores que hayan solicitado la intervención de un periférico. Este método no es posible utilizarlo más que en determinado tipo de sistemas.

Conceptos básicos

Representación de números en coma fija

Este tipo de representación sólo se utiliza para números enteros, tanto positivos como negativos. En el caso de que fuera necesario trabajar con números reales con esta representación, el programa sería el responsable de controlar los decimales. Existen, básicamente, tres formas distintas de representar un número en sistema binario con coma fija:

1. *Representación en verdadera magnitud y signo*

Este método consiste en codificar en el primer bit de la palabra el signo del número mediante un «1», en el caso de números negativos, o un «0», para los positivos, y representar con los restantes bits el valor absoluto en sistema binario. Ejemplo:

NUMERO DECIMAL	REPRESENTACION (Verdadera magnitud y signo)
4	0 0 0 0 0 1 0 0
− 4	1 0 0 0 0 1 0 0
25	0 0 0 1 1 0 0 1
− 25	1 0 0 1 1 0 0 1

2. *Representación en complemento a 1*

El bit destinado al signo, al igual que en el caso anterior, valdrá «0» ó «1», según estemos representando un número positivo o negativo, respectivamente; en el resto de los bits de la palabra se codificará la verdadera magnitud del número en binario si éste es positivo y en complemento a uno, es decir, cambiando los unos por ceros y viceversa, si es negativo. Ejemplo:

NUMERO DECIMAL	REPRESENTACION (En complemento a 1)
4	0 0 0 0 0 1 0 0
− 4	1 1 1 1 1 0 1 1
25	0 0 0 1 1 0 0 1
− 25	1 1 1 0 0 1 1 0

3. *Representación en complemento a 2*

Para representar un número en complemento a 2, basta con codificarlo en verdadera magnitud y signo si es positivo, y seguir los siguientes pasos si es negativo:

Paso 1: Representar el número en verdadera magnitud.
Paso 2: Obtener su complemento a 1.
Paso 3: Sumar una unidad a la configuración resultante. Ejemplo:

NUMERO DECIMAL	REPRESENTACION (En complemento a 2)
4	0 0 0 0 0 1 0 0
− 4	1 1 1 1 1 1 0 0
25	0 0 0 1 1 0 0 1
− 25	1 1 1 0 0 1 1 1

Justificación de las representaciones en complemento

El hecho de haber elegido dos sistemas aparentemente tan extraños para la representación de números, como son los complementos a 1 y 2 se debe a que, ambos casos, mediante un único circuito se realizan las operaciones de suma y resta, mientras que con la representación en verdadera magnitud y signo serían necesarios dos circuitos distintos. Veámoslo con un ejemplo:

• En complemento a 1:

4 − 25 = 0 0 0 0 0 1 0 0 +
+ 1 1 1 0 0 1 1 0 =
= 1 1 1 0 1 0 1 0 = − 21

4 + 25 = 0 0 0 0 0 1 0 0 +
+ 0 0 0 1 1 0 0 1 =
= 0 0 0 1 1 1 0 1 = 29

Tanto en el caso de la suma como en la resta hemos utilizado la operación (circuito) suma binaria.

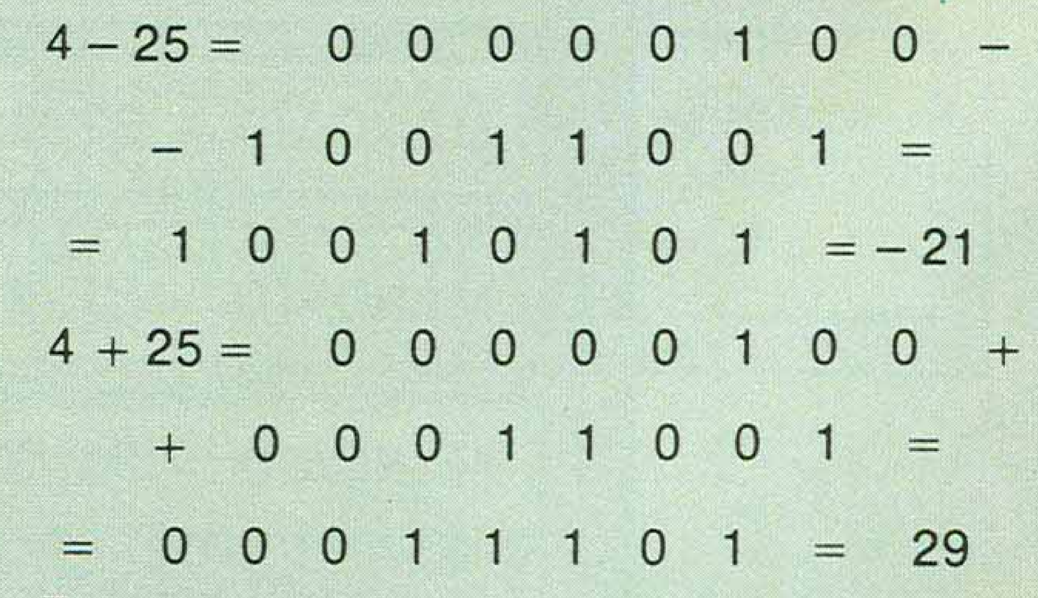
• En complemento a 2:

4 − 25 = 0 0 0 0 0 1 0 0 +
+ 1 1 1 0 0 1 1 1 =
= 1 1 1 0 1 0 1 1 = − 21

4 + 25 = 0 0 0 0 0 1 0 0 +
+ 0 0 0 1 1 0 0 1 =
= 0 0 0 1 1 1 0 1 = 29

Igual que en el caso anterior, con un único circuito «suma» podemos sumar y restar.

• En verdadera magnitud y signo:


4 − 25 = 0 0 0 0 0 1 0 0 −
− 1 0 0 1 1 0 0 1 =
= 1 0 0 1 0 1 0 1 = − 21

4 + 25 = 0 0 0 0 0 1 0 0 +
+ 0 0 0 1 1 0 0 1 =
= 0 0 0 1 1 1 0 1 = 29

En este caso, para efectuar las operaciones restar y sumar hemos necesitado dos circuitos totalmente distintos, basados en la resta y suma binaria, respectivamente.

OSBORNE 1

EL Osborne 1 fue el primer ordenador personal integrado y transportable que apareció en el mercado. Efectivamente, en la misma carcasa van incorporadas dos unidades de disquette y la pantalla. El artífice de la innovación fue Adam Osborne, conocido por sus libros y tratados en el campo del software.

La portabilidad de la máquina no fue la única novedad. A pesar de ser moderado, el precio de venta del Osborne 1, éste incluye los disquettes del sistema operativo y varios programas de aplicación.

No es de extrañar que durante los ocho primeros meses de vida en el mercado se vendiesen once mil máquinas y hubiera pedidos por más de cincuenta mil.

Ante tal éxito, pronto aparecieron los imitadores. Así, nuevos nombres como Kaipro (ahora Kaycomp), Zorba, DOT, Hyperion, etc., irrumpieron con sistemas de aspecto similar. Son las máquinas que se ha dado en llamar «clónicas» del Osborne... Y sólo han pasado dos años desde que Adam Osborne presentase su ordenador en Chicago.

Unidad central

Estamos ante otro ordenador diseñado en base al microprocesador Z-80A de Zilog. Por tanto, se trata de un sistema cuya arquitectura es de ocho bits, con un bus de direcciones de 16 bits. La memoria RAM del sistema es de 64 Kbytes, no ampliables. De los 64 K, 60 K son utilizados para los programas del usuario y los 4 K restantes se emplean para almacenar lo que aparece en la pantalla.

Cuando se carga el sistema operativo a partir del disco, éste queda residente, ocupando 12 Kbytes de la RAM. El intérprete del lenguaje BASIC ocupa 23,5 Kbytes.

La ROM del Osborne 1 tiene una capacidad de 4 Kbytes y sirve para inicializar el sistema una vez conectada la alimentación; por ejemplo, entre otras cosas, da las órdenes para cargar el sistema operativo en la memoria principal.

En la pared frontal del panel, que aparece al abatir el teclado, están dispuestas la pantalla y las bocas de acceso a las dos unidades de disco, así como dos receptáculos para guardar disquettes. Justamente debajo se observan una serie de conectores dispuestos en línea. Mirando de frente, el primer conector por la izquierda lleva la leyenda «modem». Está destinado a conectar un modulador-demodulador (modem) con acoplador acústico, para intercambiar información a través de un teléfono. A continuación, aparece un conector para interface serie, que responde a la norma RS-232. Se utiliza para conectar la impresora u otros periféricos estándar. El siguiente port (acceso) está configurado para la norma IEEE-488, muy útil para conectar la gran cantidad de instrumentos científicos de medida que utilizan dicha configuración. En realidad, el conector IEEE-488 es un borde de tarjeta de circuito impreso que permite el acceso a un conjunto de pistas del circuito impreso principal del microordenador. Y, por ser un acceso paralelo, mediante software se puede transformar fácilmente en un acceso paralelo tipo Centronics.

El siguiente conector está destinado exclusivamente al teclado. Si se desea disponer de un monitor externo habrá que recurrir al conector marcado con las palabras EXT VIDEO. Por último, existe, también, un conector que per-

Ordenador: **OSBORNE 1.**
Fabricante: **Osborne Computer.**
Nacionalidad: **Estados Unidos.**
Distribuidor en España: **Investrónica.**

CARACTERISTICAS BASICAS

UNIDAD CENTRAL	MEMORIAS DE MASA
CPU: Microprocesador Z-80A. *RAM versión básica:* 64 Kbytes (no es ampliable). *ROM versión básica:* 4 Kbytes. *Accesos periféricos:* Serie RS-232, paralelo IEEE-488 o Centronics y salida monitor.	*Discos flexibles:* Doble unidad de disco de 5 y 1/4 pulgadas de 102 Kbytes cada uno, montados en la unidad central. *Opciones:* Pueden utilizarse discos de doble densidad, con lo que almacenan 204 Kbytes cada uno.
TECLADO	**SISTEMAS OPERATIVOS**
Versión estándar: QWERTY de 69 teclas con teclado numérico.	*Estándar:* CP/M-80, 2.2.
PANTALLA	**LENGUAJES**
Versión estándar: Monocromática. *Formato de presentación:* «ventana» de 24 líneas × 52 columnas. *Opciones:* Monitor de 12″.	*En versión básica:* CBasic y MBasic. *Opcionales:* Pascal, Forth, Fortran y, en general, cualquiera que tenga versión CP/M.

mite utilizar como alimentación una batería de 12 VCC.

Teclado

Como hemos visto anteriormente el teclado es abatible. De hecho constituye la tapa que protege al panel frontal del Osborne. Por su parte exterior lleva el asa que sirve para transportar la máquina.
Un total de 69 teclas conforman el teclado. De ellas, 57 componen el teclado principal. Su configuración es del tipo QWERTY, con posibilidad de letras mayúsculas y minúsculas. Además, existen cuatro teclas para el movimiento del cursor en la pantalla.
Las doce teclas restantes forman el teclado numérico, que consta de los diez números del 0 al 9, el punto decimal y la tecla «return».

Pantalla

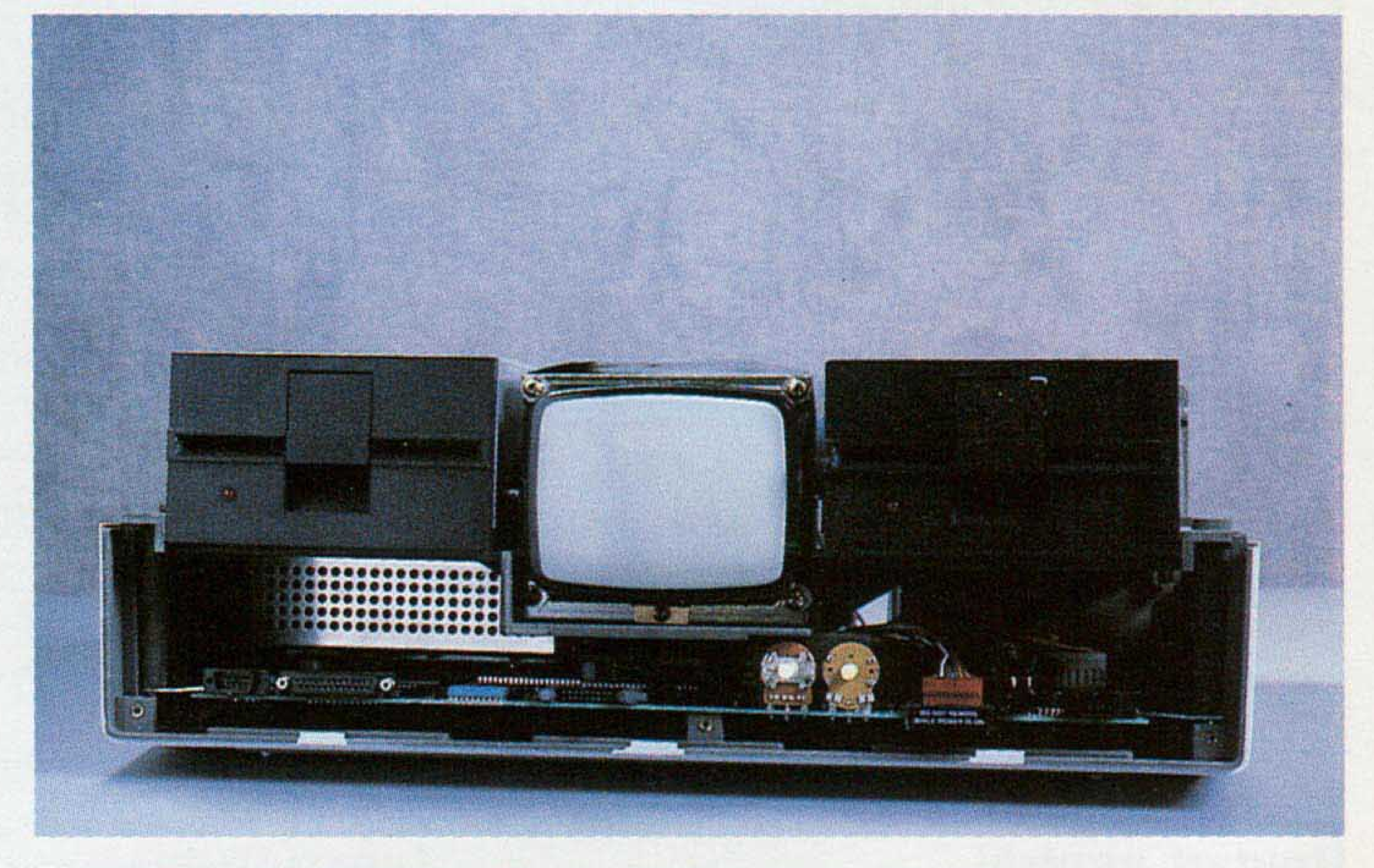
La pantalla es monócroma. Los caracteres aparecen en verde o blanco (la versión para Europa es verde) sobre fondo oscuro, en un TRC (Tubo de Rayos Catódicos) de 5 pulgadas.
El formato real de la pantalla es de 32 líneas por 128 caracteres, si bien, en el monitor sólo aparece una ventana de 24 líneas de 52 caracteres, que se desplaza automáticamente o a voluntad por toda la página (scrolling vertical y horizontal).
Los caracteres son formados por una matriz de ocho por diez puntos. Se dispone de un conjunto de 96 caracteres ASCII, minúsculas acentuadas y 32 caracteres gráficos. Todos estos caracteres se pueden subrayar.

Memorias de masa

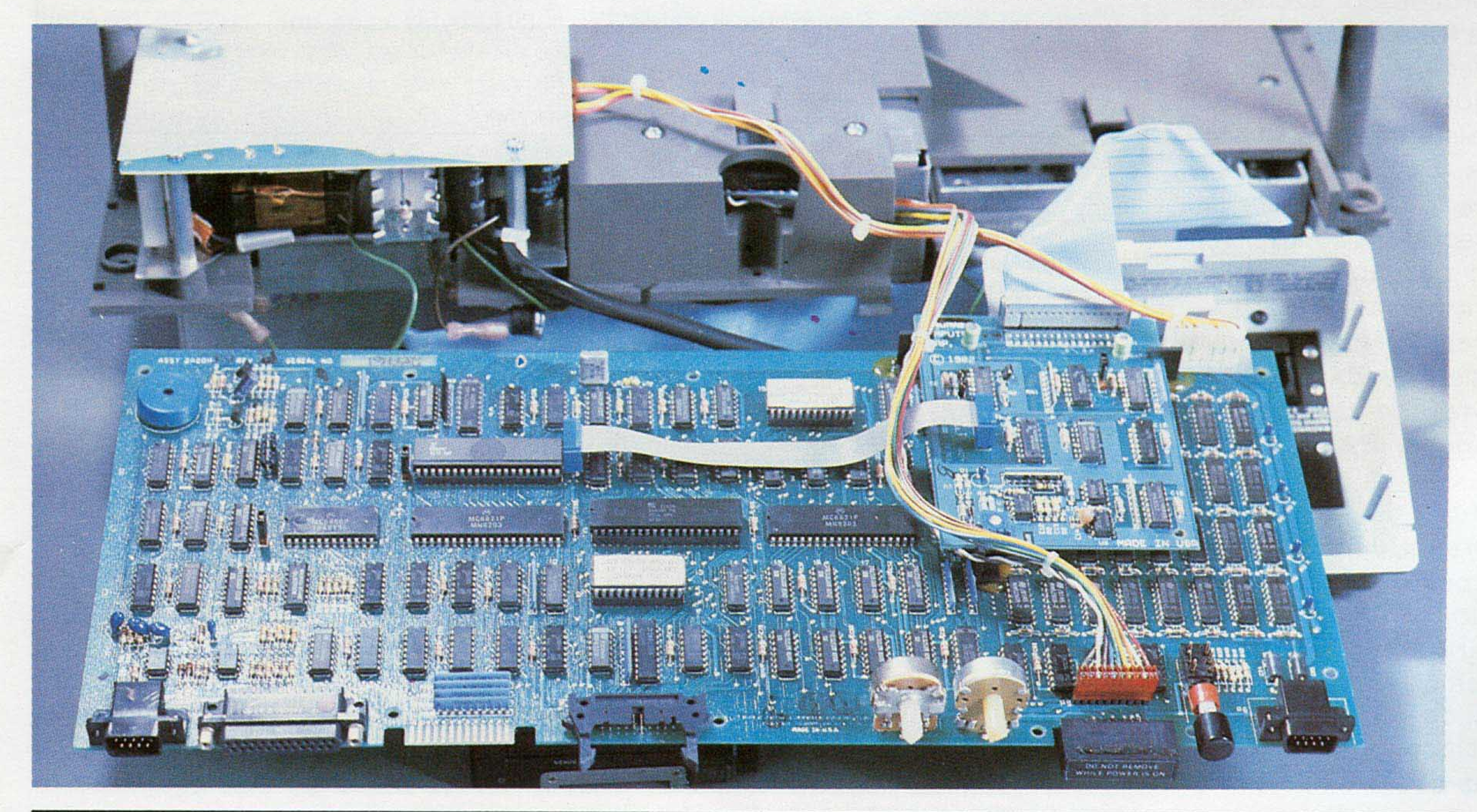
Según vimos anteriormente, el Osborne 1 viene provisto de dos unidades de disco situadas en la parte frontal del sistema. Cada una de ellas puede alma-



El OSBORNE-1 es el primer ordenador personal integrado y transportable que apareció en el mercado.



El equipo está integrado en un maletín en cuya tapa superior se encuentra el teclado. El resto del mueble aloja a la unidad central, la pantalla de visualización y dos unidades de disco flexible.



La CPU del OSBORNE-1 está constituida por el microprocesador de 8 bits Z-80 A de la firma Zilog. La zona de memoria RAM se eleva a 64 Kbytes ya incorporados en la versión básica.

cenar hasta 102 Kbytes en un disquette de 5 1/4 pulgadas. Opcionalmente, se puede disponer de unidades de doble densidad, con lo que se duplica la capacidad de almacenamiento (hasta 204 Kbytes por disco).
No existe posibilidad de conectar un mayor número de unidades de disco.

Periféricos

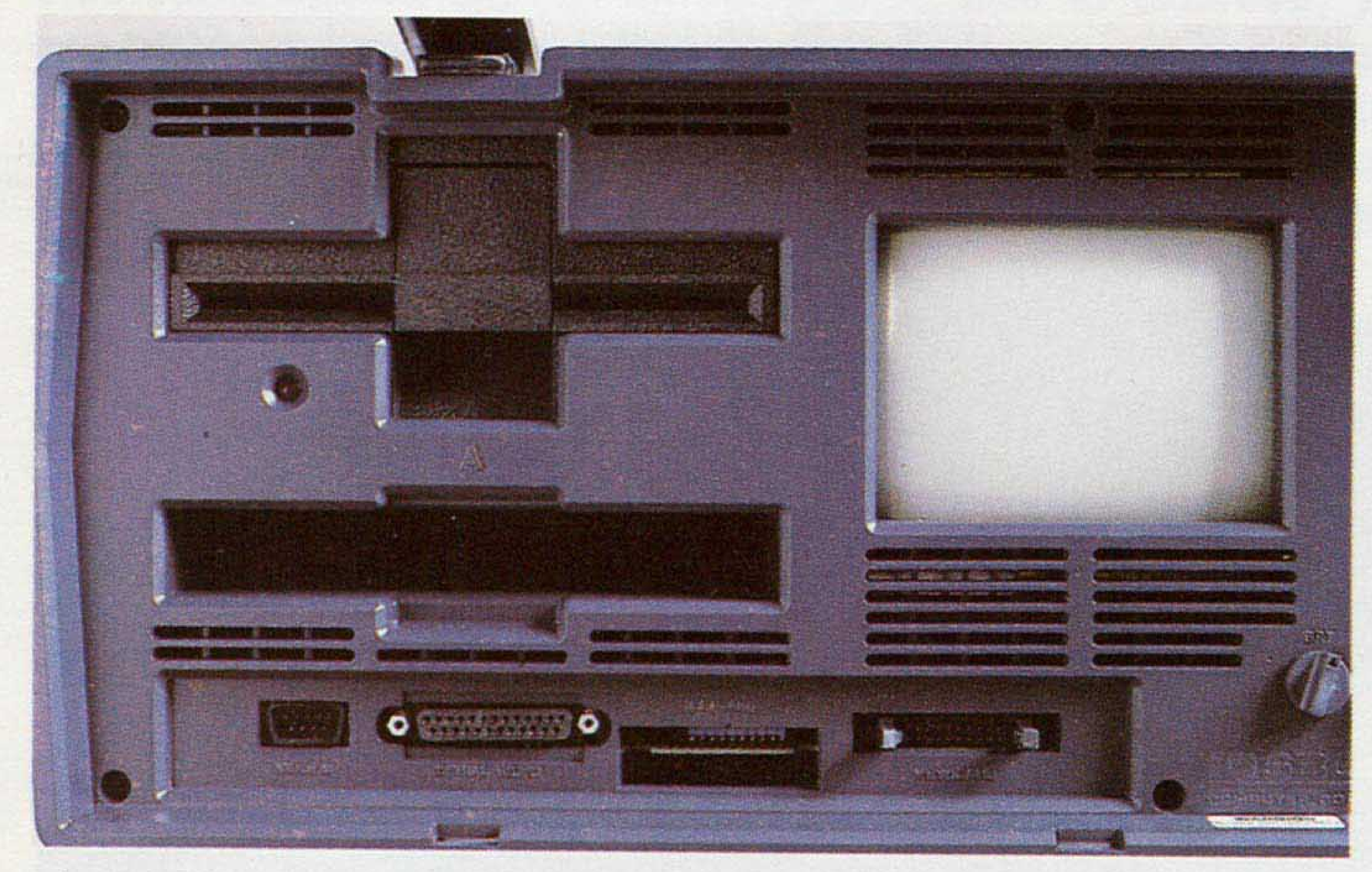
En primer lugar, se puede utilizar un monitor de 12 o más pulgadas siempre que se crea conveniente. Sin embargo, el periférico más útil casi siempre resulta ser la impresora. En el caso que nos ocupa se puede emplear casi cualquier tipo de las existentes en el mercado, a través tanto del interface RS-232 como del Centronics. Obviamente, pueden utilizarse tanto impresoras de matriz de puntos como de margarita, si se desea una mayor calidad de escritura.

Sistemas operativos y lenguajes

Esta máquina entra de lleno en la clasificación de microordenadores que han dado en llamarse CP/M, por el sistema operativo que utilizan para su gobierno. Por el momento el fabricante no piensa en ningún otro sistema operativo.
Los lenguajes de alto nivel que pueden utilizarse para programar el Osborne 1, son tanto el CBasic (Basic compilado) como el MBasic (Basic interpretado). Opcionalmente cabe utilizar el Pascal, el Forth y el Fortran o, en general, cualquier lenguaje que tenga versión para sistema operativo CPM.

Software de aplicación

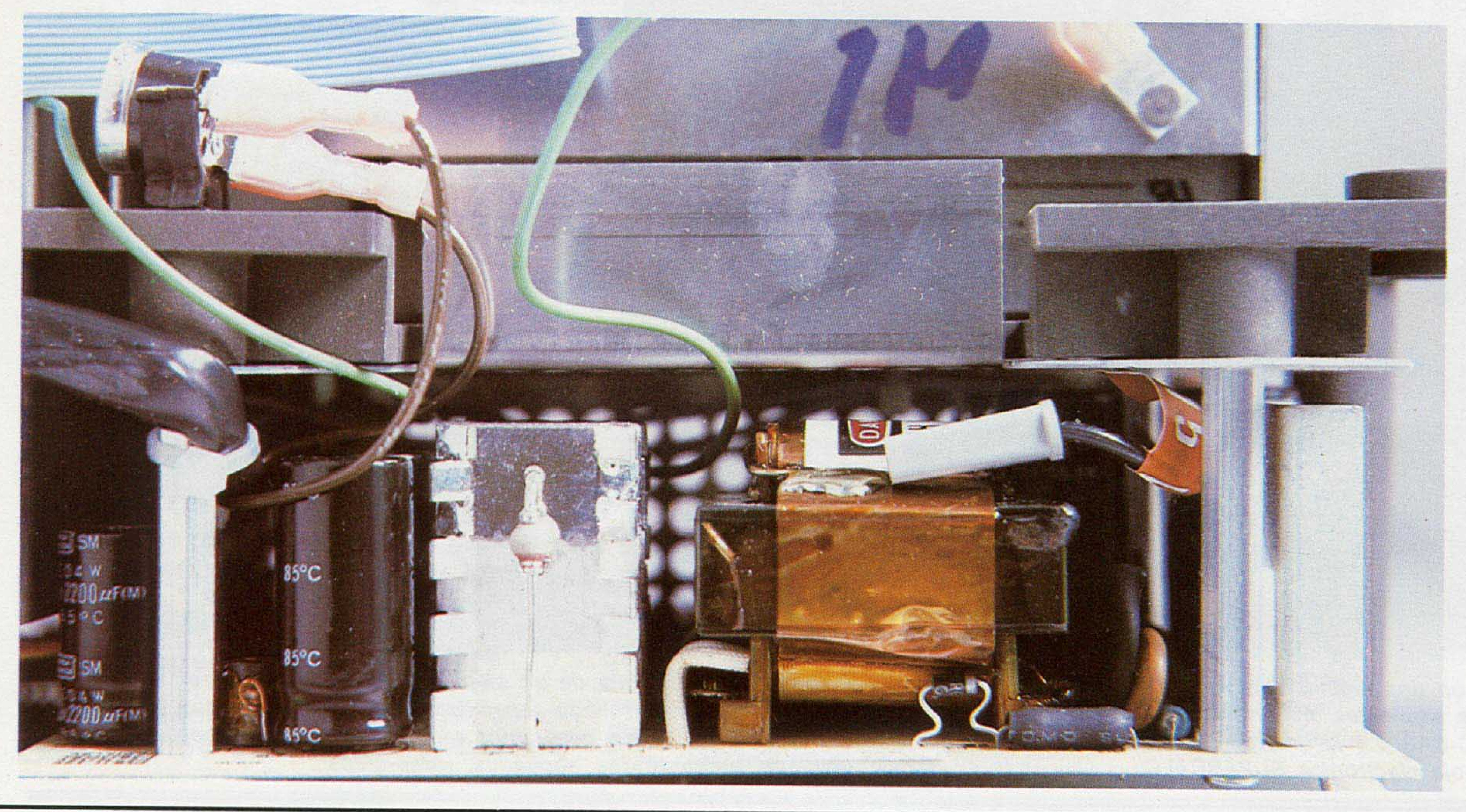
En el precio de compra se incluyen, además del sistema operativo (CP/M 2.2) y los lenguajes CBasic y MBasic, el paquete para tratamiento de textos Wordstar, acompañado por el Mailmerge y la hoja de trabajo Supercalac para trabajos financieros.



La minipantalla de 5 pulgadas incorporada al equipo es monócroma de fósforo verde en la versión europea. La presentación corresponde a una ventana de 24 líneas de 52 caracteres que se desplaza a través de la página de visualización.



Debajo de las bocas de acceso a las dos unidades de disco existen sendos receptáculos para almacenar los discos flexibles. Los conectores para comunicaciones están situados justamente debajo de los citados receptáculos.



Su naturaleza compacta es la característica más significativa del OSBORNE-1. Además, de la propia fuente de alimentación que le permite operar conectado a la red eléctrica, el equipo puede alimentarse a partir de una batería de 12 V.

OSBORNE 1

Un paquete interesante que puede utilizarse con el Osborne es la base de datos DBASE II. Lógicamente el ordenador puede utilizar la mayoría de los programas existentes en el mercado y que están escritos para sistema operativo CP/M.

El distribuidor dispone de una amplia biblioteca de programas desarrollados específicamente para el Osborne, entre los que destacan los desarrollados para la ayuda en su gestión a:

— Médicos.
— Asentadores de frutas y verduras.
— Procuradores de Tribunales.
— Agencias de transportes.
— Abogados.
— Video-clubs.
— Odontólogos.
— etc.

Algunos de los cuales son tratados en las páginas de esta enciclopedia.

Soporte y distribución

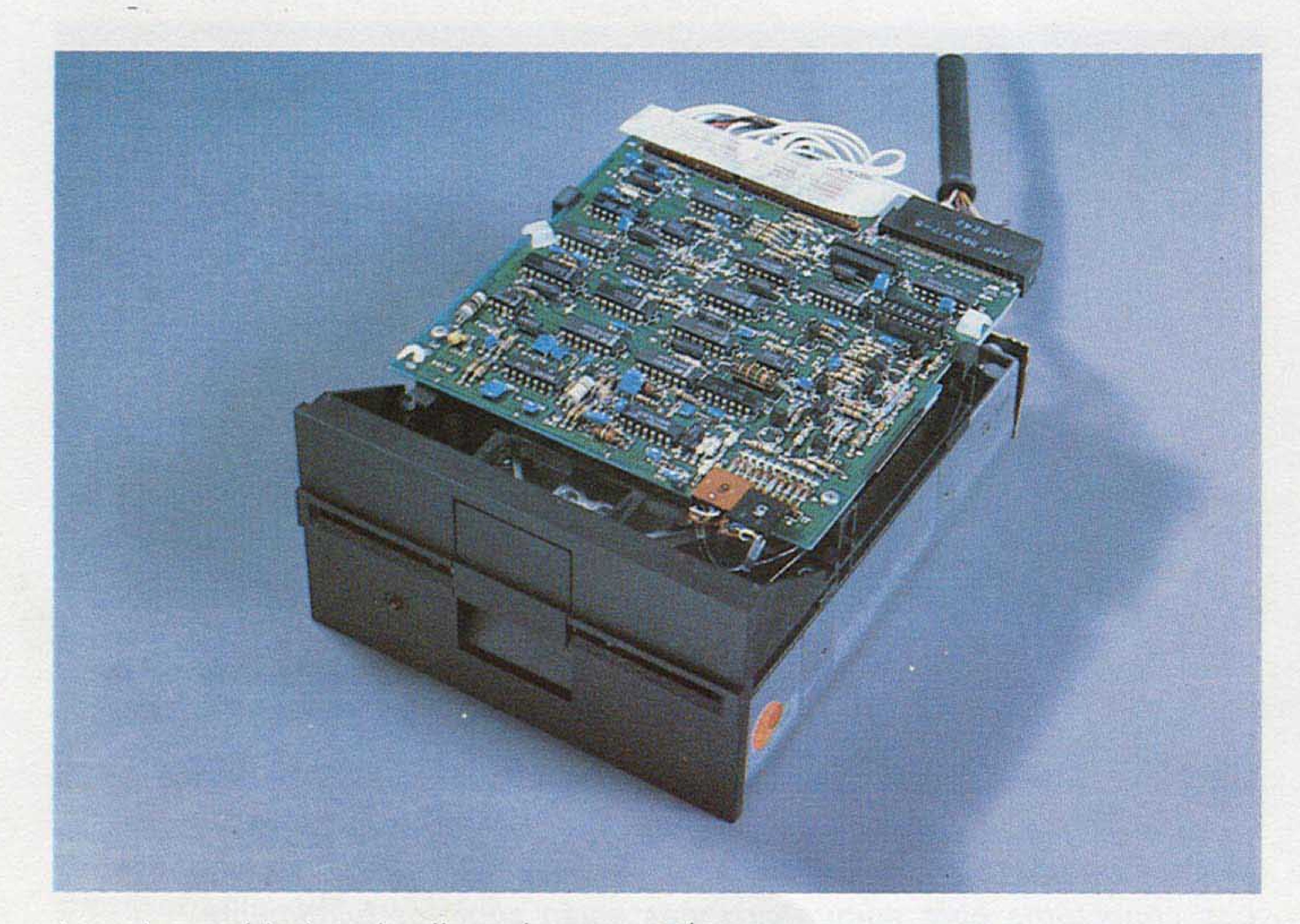
Cada ordenador Osborne 1 se entrega al cliente junto con los disquettes de software antes indicados y una amplia información redactada en castellano, que incluye en un sólo tomo: una guía del usuario, un Manual de los lenguajes, un Manual CP/M, un Manual Wordstar y un Manual Supercalc.

El sistema se comercializa a través de la amplia red de distribuidores de que dispone Investrónica que es el importador exclusivo para España.

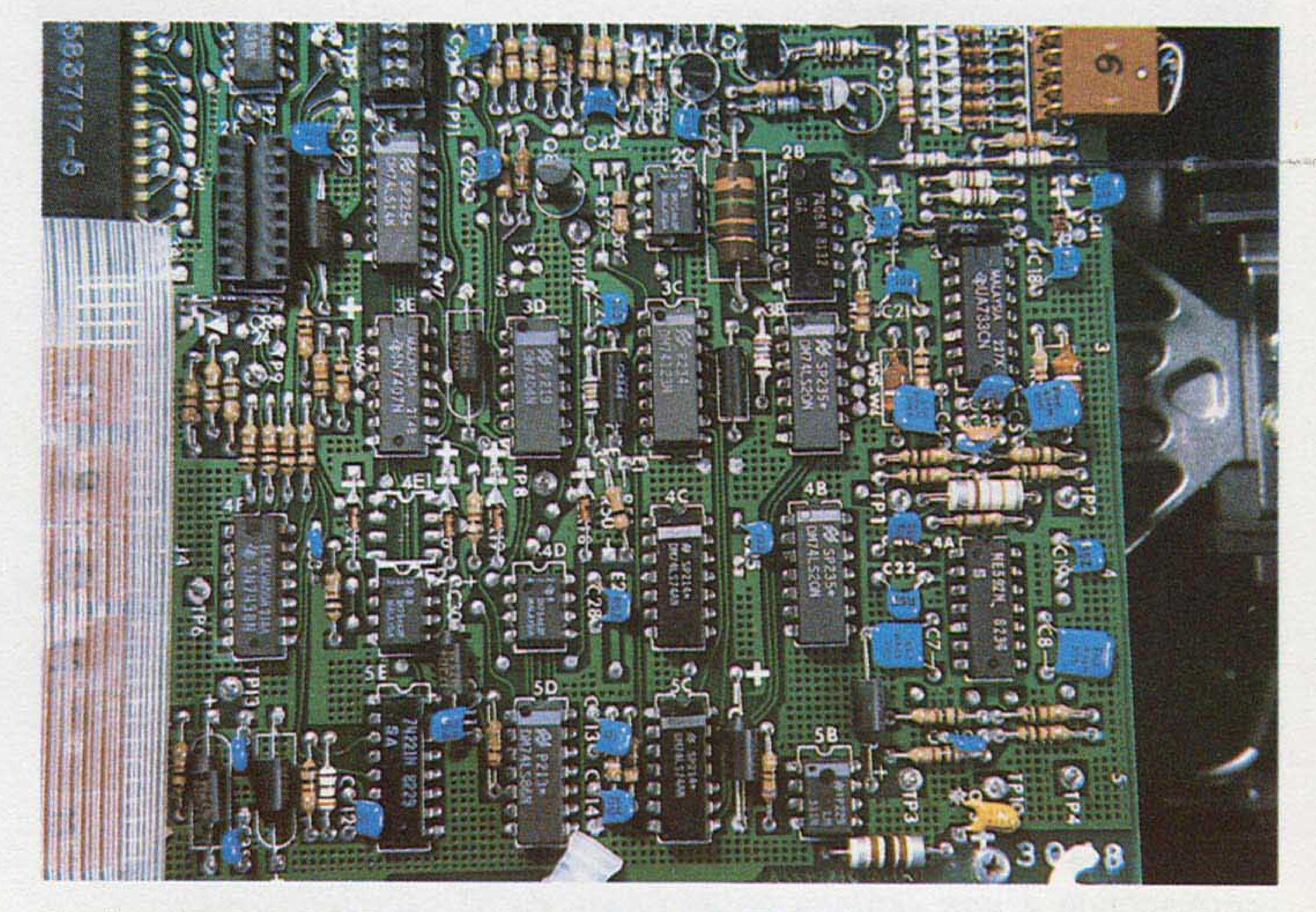
Los equipos se entregan con una garantía de tres meses, al término de la cual existe la posibilidad de un contrato de mantenimiento.

Configuración básica mínima. Unidad central (64 Kbytes de RAM), teclado español, monitor de 5″ incorporado, dos unidades de disco de 102 Kbytes (también incorporados) e impresora.

Configuración máxima. Unidad central de 64 Kbytes de RAM, teclado español, monitor incorporado y monitor auxiliar de 12″ o más, dos unidades de disco de 204 Kbytes e impresora.



Las dos unidades de disco incorporadas al equipo son para discos de 5 y 1/4 pulgadas con una capacidad de 102 Kbytes por disco.



Opcionalmente, pueden sustituirse las unidades de disco que incorpora la versión estándar por unidades para discos de doble densidad con capacidad de 204 Kbytes por unidad.



El equipo se entrega con una amplia documentación redactada en castellano y con los siguientes disquettes: sistema operativo (CP/M 2.2), lenguajes C-BASIC y M-BASIC, tratamiento de textos WORDSTAR y hoja electrónica SUPERCALC.



El distribuidor dispone de un amplio catálogo de software de aplicación desarrollado específicamente para el equipo; la mayor parte de los programas están orientados a la gestión: médicos, procuradores, abogados, video-clubs...

LOS LENGUAJES MAQUINA

COMO ya sabemos, cualquier ordenador dispone de un conjunto o «repertorio» de instrucciones elementales en «código de máquina», que le indican lo que tiene que hacer.
Cada instrucción debe contener diversos elementos de información, con el fin de que el ordenador la pueda interpretar y, en consecuencia, ejecutar. Una de estas informaciones es lo que llamamos *código de operación,* que indica a la unidad de control cuál es la operación que debe efectuar. El resto de la instrucción (operando) debe indicar el/los dato/s o la dirección de la posición de memoria en que se encuentra el dato o los datos con los que se va a efectuar la operación. Por último, a veces es necesaria una información complementaria (de dispositivo de E/S, *status,* etc.) y que denominaremos *indicador.*

Formatos de las instrucciones

Las instrucciones de las diversas máquinas tienen un formato muy variado, siendo quizá la más compleja la del tipo:

CO	D10	D20	DR	DSI	I

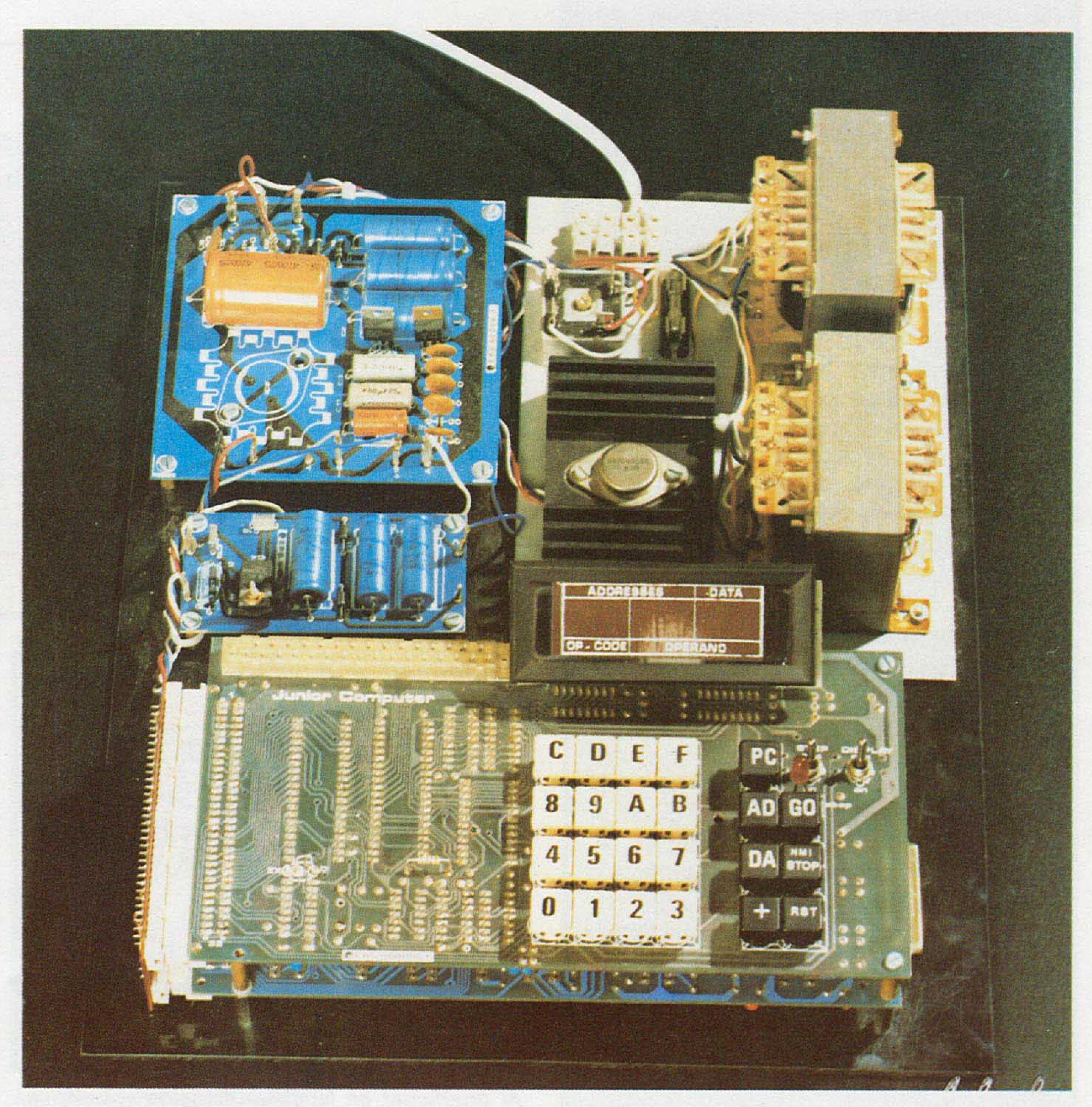
en la que CO es el código de operación; D10, la dirección del primer operando; D20, la dirección del segundo operando; DR, la dirección en la que se debe almacenar el resultado; DSI, la dirección en la que se encuentra la siguiente operación, y, por último, I un indicador que especifica —cuando existe— algo más sobre la instrucción en concreto (periférico, registro especial, etc.).
Este caso sería el de una instrucción de cuatro direcciones, aunque las hay también de tres, siendo lo normal que sean de dos o de una, dependiendo de la arquitectura de la CPU.
En los microordenadores, las instrucciones de nivel máquina suelen ser de uno, dos o tres bytes. El primer byte contiene el código de operación, mientras que los otros contienen el dato, dirección del dato o el indicador.

¿Cómo opera el procesador?

El procesador (microprocesador en el caso de los sistemas microordenadores) dispone de registros para la ejecución del programa: el contador de programa (en donde se almacena la dirección de la próxima instrucción que se debe ejecutar) y el registro de instrucción (en donde se decodifica la instrucción). El contador de programa indica en qué dirección de la memoria se encuentra almacenada la instrucción. Al leer ésta, el primer byte que contiene el código de operación es trasladado, a través del bus de datos, al registro de instrucción, donde es interpretado por el decodificador; acto seguido, el contador de programa se incrementa en una unidad. Si la instrucción contiene más bytes, el segundo pasa al registro de instrucción y se repite el proceso; y así sucesivamente. Una vez que se han decodificado o interpretado todos los bytes, se ejecuta la instrucción. El proceso se repite para la instrucción siguiente: se traslada el nuevo código de operación al registro de instrucción, desde la dirección indicada por el contador, repitiéndose el proceso hasta que se llega al final del programa.

¿Programar en binario?

Dado que todo el proceso se realiza con bits, es natural que, en principio, el programa deba estar en lenguaje binario. El programador debe escribir en binario tanto el código de operación como los datos y direcciones, tal como era preciso en los primeros ordenadores. El siguiente paso consiste en utilizar el sistema hexadecimal o decimal para escribir las instrucciones. Así, se simplifica la labor del programador. Los códigos de operación y las direcciones en decimal se utilizaron en algunos ordenadores de la segunda generación, aunque hoy día lo usual es



Los microordenadores menos evolucionados suelen programarse en el lenguaje máquina «numérico» propio del microprocesador que constituye su CPU. La representación numérica de la información suele realizarse en el sistema hexadecimal.

LOS LENGUAJES MAQUINA

Conceptos básicos

Introducción a la teoría de los lenguajes

Noam Chomsky inició el estudio de los lenguajes formales al crear un modelo matemático de una gramática en 1956. Hoy día el estudio de las gramáticas formales es uno de los campos más importantes de la informática teórica.

Webster define el lenguaje como «el conjunto de palabras y reglas para combinarlas, que es usado y entendido por una comunidad numerosa». Pero esta definición clásica no es rigurosa matemáticamente. Cuando los seres humanos usan un lenguaje, se puede permitir la existencia de palabras, términos o frases con significado simbólico o ambiguo, ya que la inteligencia y el buen juicio del oyente le permite comprender el sentido exacto que quiere darle el emisor, soslayando las discrepancias que puedan existir entre la forma y el fondo, esto es, entre la sintaxis y la semántica.

Para que una máquina entienda el significado concreto de un mensaje necesita que se le comunique en un lenguaje que esté rigurosamente definido con unas estrictas reglas gramaticales.

La comunicación hombre-máquina se debe realizar a través de un lenguaje escrito que impida la posibilidad de error en la interpretación de los mensajes.

Los conceptos fundamentales de la teoría de lenguajes son los de alfabeto, cadena, lenguaje y gramática.

Se llama *alfabeto* a un conjunto no vacío de símbolos gráficos.

Ejemplos típicos son:

El alfabeto castellano = {ⓐⓑⓒ ... ⓩ}

El alfabeto griego = {(α)(β)(γ) ... (ω)}

El alfabeto binario = {⓪①}

Hemos encerrado los elementos del alfabeto en círculos para distinguirlos, ya que ni la coma ni el espacio en blanco indicarían la separación, puesto que éstos pueden ser elementos del lenguaje (el espacio o la coma tienen ese carácter en muchos lenguajes de ordenador). No obstante, lo normal es representar los elementos espaciados o separados por comas. A los alfabetos se les suele llamar también *conjuntos de base* o *vocabularios* y se les representa por los símbolos Σ o V. Hay alfabetos que no son gráficos, como el alfabeto musical, formado por las notas (aunque luego se las represente gráficamente mediante símbolos en un pentagrama), el de los sordomudos o el de los ciegos.

utilizar la representación hexadecimal. Estos lenguajes de máquina reciben el calificativo de *numéricos* para distinguirlos de los lenguajes de máquina *simbólicos*.

Estos últimos sustituyen el código de operación numérico por un código alfabético que es nemotécnico, es decir, que le recuerda al operador lo que tiene que hacer la instrucción. Es más fácil recordar que ADD significa sumar (sobre todo, si se tienen conocimientos elementales de inglés), que mantener en la memoria que para sumar hay que emplear el código hexadecimal 09 (adición en el microprocesador Z80).

Siempre que se programe en el lenguaje propio de la máquina el programador debe llevar el control de las posiciones de memoria en las que almacena los diferentes datos, es decir, necesita utilizar el llamado mapa de direcciones de memoria.

Tipos de instrucciones

Las instrucciones se suelen agrupar en tipos que determinan la naturaleza de la tarea que se ordena a la CPU. Los principales tipos son:

- *Instrucciones de transferencia de datos.* Permiten la lectura o escritura desde o hacia la memoria, y entre registros internos del procesador. Incluyen también la carga y descarga de registros (de memoria a acumulador o viceversa, etc.).
- *Instrucciones de ruptura de secuencia.* Son las instrucciones que realizan los saltos y las bifurcaciones de una parte a otra del programa. Pertenecen a este grupo tanto los saltos incondicionales como los condicionales, las instrucciones de control de bucles, las llamadas a subrutinas, las de retorno al programa principal, etc.

Instrucción de un byte	B7 B6 B5 B4 B3 B2 B1 B0	
Instrucción de dos bytes	B7 B6 B5 B4 B3 B2 B1 B0	D7 D6 D5 D4 D3 D2 D1 D0
Instrucción de tres bytes	B7 B6 B5 B4 B3 B2 B1 B0	D7 D6 D5 D4 D3 D2 D1 D0 D7 D6 D5 D4 D3 D2 D1 D0
	Código operación	Dato o dirección

En general, existen tres formatos de instrucciones a nivel máquina: de uno, dos o tres bytes. El primer byte corresponde al código de operación y los restantes contienen el operando (dato o dirección del dato).



Al programar en lenguaje máquina «numérico» en su nivel más elemental (sistema binario), el programador debe escribir las instrucciones en lenguaje binario: ceros y unos.



Con un leve perfeccionamiento, consistente en la inclusión de un codificador, la tarea de programación en lenguaje máquina puede facilitarse al permitir la escritura de las instrucciones en hexadecimal.

- *Instrucciones de entrada/salida.* Las que relacionan a los periféricos con la memoria.
- *Instrucciones de control.* Las que permiten controlar el programa y el equipo. Comprenden el control del *status* (registro de estado), la no operación, la parada, etc.

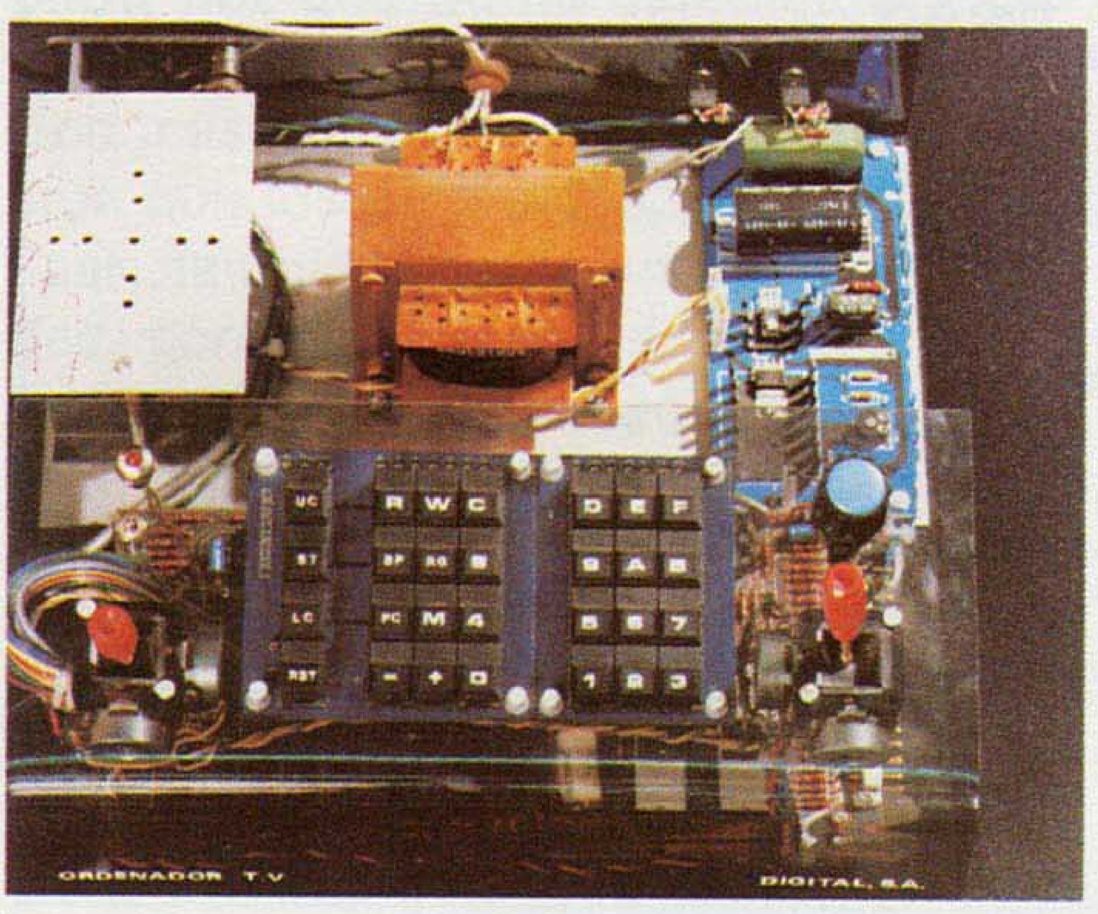
Los conjuntos de instrucciones máquina, así como sus códigos nemotécnicos difieren de un equipo a otro, por lo que es necesario que el programador disponga de la tabla correspondiente.

Direccionamientos

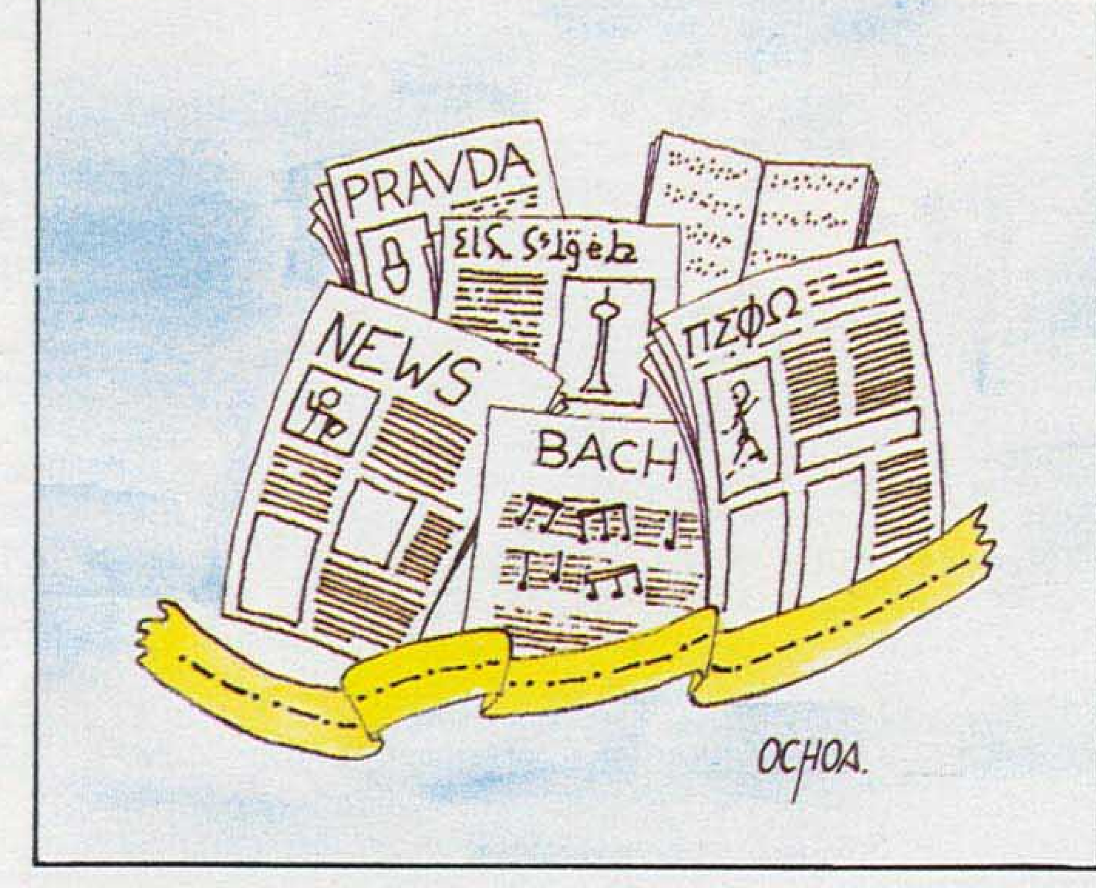
Los bytes que siguen al código de operación corresponden al operando: un dato, una dirección o un indicador. El código de operación indica a la unidad de control el tipo de direccionamiento implicado y, por consiguiente, dónde localizar el dato correspondiente. Los tipos de direccionamientos más usuales son:

- *Direccionamiento implícito:* Opera entre registros internos. Son instrucciones de un sólo byte. Ejemplo: RTS (60) del microprocesador 6502 que corresponde retorno de subrutina.
- *Direccionamiento inmediato:* Opera directamente con el dato contenido en los bytes de operando. Ejemplo: ADI (C5) del 8080. Suma al acumulador el segundo byte de la instrucción.
- *Direccionamiento directo:* Accede a la posición indicada por los bytes de operando. En esa posición de memoria se encuentra el dato a operar. Ejemplo: SUB del 8085. Resta del acumulador el contenido del registro cuya dirección está en los bytes de dirección.
- *Direccionamiento relativo:* Permite acceder a la posición de memoria, cuya

En los lenguajes de máquina simbólicos, el programador utiliza códigos nemotécnicos en lugar de códigos numéricos. En cualquier caso, sigue siendo necesario llevar un mapa de direcciones reales en las que se van almacenando los datos.



Una de las dificultades inherentes a la programación en lenguaje máquina es que los programas sólo son ejecutables en equipos análogos, basados en la misma unidad central de proceso.



Los innumerables alfabetos que existen sirven para representar de forma simbólica los lenguajes que permiten la comunicación.

Glosario

¿Es necesario saber inglés para programar ordenadores?

No, aunque sí es conveniente, ya que todos los códigos nemotécnicos de los lenguajes de programación son recordatorios del verbo inglés que indica la operación a ejecutar. Así, ADD es sumar o SUB es la abreviatura de Substract (restar). Además, los manuales de los fabricantes y mucha de la literatura informática se encuentra en inglés.

¿Por qué se suelen utilizar dos bytes para expresar direcciones de memoria en los microordenadores?

El motivo es que con 16 bits se puede representar en binario hasta el número 2^{16} = 65.536 = 64 K que coincide con el número máximo de posiciones de memoria directamente direccionables por un microprocesador de 8 bits (con bus de direcciones de 16 bits). Estos microprocesadores constituyen la CPU de la mayor parte de los microordenadores actuales.

¿Cómo se organiza el mapa de direcciones?

El mapa se suele dividir en páginas. Una página es un bloque de direcciones de memoria, usualmente de 4 Kbytes, aunque puede ser de cualquier otra potencia de 2. Así, una memoria de 64 K tendría 16 páginas de 4 Kbytes numeradas en hexadecimal de 0 a F. Cada página tendría las direcciones comprendidas entre el 000 y el FFF. De esta forma la página 0 comprendería las direcciones absolutas desde la 0000 a la 0FFF. La página 1 desde 1000 a 1FFF, y así sucesivamente. La última página, la F, abarcaría desde la dirección F000 a la FFFF.

LOS LENGUAJES MAQUINA

dirección es la indicada por el contador del programa más o menos un número indicado por el segundo byte. Ejemplo: BCC (90) del 6502. Salta a la dirección del contador más el valor del segundo byte, atendiendo a un determinado indicador del registro de estado.

- *Direccionamiento indexado:* La dirección del dato se obtiene sumando el valor del registro indicado al valor de la dirección absoluta incluida en la instrucción. Ejemplo: DEC (D6) del 6502, que resta una unidad al valor que está en la dirección dada por el contenido del registro índice X más el valor del segundo byte de la instrucción.

Al igual que ocurre con los repertorios de instrucciones máquina, los tipos de direccionamiento de las instrucciones difieren de un equipo a otro y son función del diseño de la CPU.

¿Hay ventajas en programar en lenguaje de máquina?

El conocimiento exhaustivo de las instrucciones (códigos y forma de direccionamiento), así como de las direcciones reales de memoria en donde se almacenan los datos, es una de las principales dificultades para programar en lenguajes de máquina. Además, los programas en código de máquina no son ejecutables en otro equipo que no tenga un procesador igual o compatible.

Entonces, ¿es útil conocer el lenguaje máquina de nuestro equipo, aunque éste admita la programación en un lenguaje de alto nivel? La respuesta es sí, por las siguientes razones:

— Si tenemos poca capacidad de memoria podremos recurrir al lenguaje de máquina que es el que menos memoria ocupa.

— Una rutina que se haya de utilizar en múltiples ocasiones, programada en código de máquina no sólo ahorra memoria, sino también tiempo de ejecución.

— Las mejoras o modificaciones en el sistema operativo son más efectivas si se hacen en lenguaje de máquina.

Por último, recordemos que los programas en código de máquina son los que se ejecutan en menos tiempo.

```
                      REPEAT ROUTINE
0000                        ORG   $0000
                      TEMPORARY DATABUFFERS IN PAGE ZERO
0900                  KEY    *     $00DA
0000                  NOTEL  *     $00DC
0000                  NOTEH  *     $00DD
0000                  LENGTH *     $00DE
                      INTERVAL TIMER
0000                  CNTA   *     $1AF4  DISABLE TIMER IRQ
0000                  CNTD   *     $1AF7  DISABLE TIMER IRQ, CLK1KT
0000                  CNTG   *     $1AFE  ENABLE TIMER IRQ, CLK64T
0000                  RDFLAG *     $1AD5  B7 IS TIMER FLAG
                      GOTO MONITOR
0000                  RESET  *     $1C1D  NEW I/O DEFINITION
                      I/O DEFINITION
0000                  PBD    *     $1A82
0000                  PBDD   *     $1A83
                      IRQ VECTOR
0000                  IRQL   *     $1A7E
0000                  IRQH   *     $1A7F
                      START OF THE REPEAT PROGRAM
0000 78               REPEAT SEI          DISABLE IRQ LINE
0001 D8                      CLD
0002 A9 30                   LDAIM IRQRE  SET UP IRQ VECTOR
0004 8D 7E 1A                STA   IRQL
0007 A9 1A                   LDAIM IRQRE  /256
0009 8D 7F 1A                STA   IRQH
000C A9 01                   LDAIM $01    PB0 IS OUTPUT
000E 8D 83 1A                STA   PBDD
0011 8D 82 1A                STA   PBD    TOGGLE SPEAKER OFF
0014 85 DD                   STAZ  NOTEH  SET NOTE POINTER
0016 A9 00                   LDAIM $00
0018 85 DC                   STAZ  NOTEL  SET NOTE POINTER
001A 8D F4 1A                STA   CNTA   RESET IRQ LINE, DISABLE TIMER IRQ
001D 58                      CLI          ENABLE CPU IRQ
001E A9 FF            FETCH  LDAIM $FF    SET TIMER ENABLE TIMER IRQ
0020 8D FE 1A                STA   CNTG
0023 A0 00                   LDYIM $00    FETCH NOTE
0025 B1 DC                   LDAIY NOTEL
0027 85 DA                   STAZ  KEY
0029 C8                      INY          FETCH LENGTH
002A B1 DC                   LDAIY NOTEL
002C 85 DE                   STAZ  LENGTH
002E A4 DA                   LDYZ  KEY    LOOKUP CONVERSION
0030 A9 00            TONE   LDAIM $00    TOGGLE SPEAKER ON
0032 8D 82 1A                STA   PBD
0035 BE 00 1A                LDXY  DEL    GET FREQUENCY
0038 20 70 00         TONEA  JSR   EQUALA DELAY 22 MICRO SEC
003B CA                      DEX
003C D0 FA                   BNE   TONEA  LOOP TIME IS 27 MIKRO SEC*X
003E A9 01                   LDAIM $01    TOGGLE SPEAKER OFF
0040 8D 82 1A                STA   PBD
0043 BE 00 1A                LDXY  DEL    GET FREQUENCY AGAIN
0046 A5 DE            TONEB  LDAZ  LENGTH GET LENGTH
0048 30 08                   BMI   TONEC  TIME OUT?
004A 20 74 00                JSR   EQUALB EQUALIZE 17 MICRO SEC
004D CA                      DEX
004E D0 F6                   BNE   TONEB  LOOP TIME IS 27 MICRO SEC*X AGAIN
0050 F0 DE                   BEQ   TONE   RETURN AFTER ONE PERIODE
0052 A2 04            TONEC  LDXIM $04    LOOP TIME = 4*CNTD*PRESET
0054 A9 30            TONED  LDAIM $30    PRESET = $30
0056 8D F7 1A                STA   CNTD   DISABLE TIMER IRQ
0059 2C D5 1A         POLL   BIT   RDFLAG READ FLAG REGISTER, TIME OUT?
005C 10 FB                   BPL   POLL   IS TIMER FLAG STILL ZERO?
005E CA                      DEX
005F D0 F3                   BNE   TONED  LOOP COUNTER ZERO?
0061 E6 DC                   INCZ  NOTEL  ADJUST NOTE POINTER
0063 E6 DC                   INCZ  NOTEL
0065 A0 00                   LDYIM $00
0067 B1 DC                   LDAIY NOTEL  END OF NOTE BUFFER?
0069 C9 77                   CMPIM $77    EOF CHARACTER
006B D0 B1                   BNE   FETCH  IF NOT EOF, CONTINUE
006D 4C 1D 1C                JMP   RESET  ELSE BACK TO MONITOR
```

Listado de un programa confeccionado en lenguaje máquina para el microprocesador 6502. En el mismo aparecen las instrucciones en lenguaje máquina numérico —hexadecimal— (zona izquierda) y en lenguaje simbólico (zona derecha).

PLOTTERS

LOS plotters son periféricos de salida que efectúan dibujos de trazo continuo al recibir las instrucciones correspondientes de un ordenador; o dicho de otro modo, a partir de un programa un plotter puede realizar los planos que corresponden a un diseño. Su aplicación principal es en oficinas de ingeniería como elemento final de salida «hard copy» (copia impresa) de los sistemas CAD (Computer Aided Design: diseño ayudado por ordenador) o CAM (Computer Aided Manufacturing: fabricación ayudada por ordenador). Si, por ejemplo, se quiere realizar el diseño de una estructura, el sistema deberá disponer de un teclado y de una pantalla de rayos catódicos con posibilidad de gráficos; mediante el teclado se realizan los cálculos correspondientes, así como las diferentes correcciones en el dibujo de la estructura que aparece en la pantalla. Una vez que ya se tiene en la pantalla el dibujo final corregido, se pasa al papel dibujándolo mediante el plotter. De igual manera se puede utilizar en electrónica para diseño de circuitos impresos: el plotter dibuja el plano del circuito impreso, el plano de montaje de los componentes y toda la información necesaria para la realización práctica del diseño electrónico.

Funcionamiento

Por la forma de realizar el dibujo los plotters se pueden dividir en dos tipos:

- *De plumas:* Los dibujos se efectúan mediante plumas con tinta que se aplica sobre un papel normal.
- *Electrostáticos:* La pluma se reemplaza por una punta catódica y se dibuja sobre un papel electrosensitivo. Son más rápidos, pero de menor precisión que los de plumas. Se pueden utilizar, también, como impresoras rápidas, con velocidades de escritura que llegan a 1.625 líneas por minuto.

Los plotters que utilizan plumas con tinta pueden ser de dos tipos:

a) *De mesa:* el tamaño del papel es normalmente DIN A-3 o DIN A-4. Este se fija por efecto electrostático o mediante regletas imantadas. La pluma se desplaza por una guía o carro que a su vez es capaz de moverse en la dirección perpendicular sobre otras guías. La mesa puede ser horizontal (flatbed) o inclinada (beltbed).

b) *De tambor:* las plumas se desplazan a lo largo de la generatriz de un cilindro en el cual se enrolla el papel. Al mismo tiempo este cilindro o tambor puede girar en uno u otro sentido mediante un motor paso a paso. Se emplea el papel en rollo y, normalmente, permiten realizar dibujos de mayor tamaño que los plotters de mesa.

Características

Las características más importantes a la hora de evaluar un plotter son:

— *Paso incremental:* Debido a que el desplazamiento de las plumas por el papel se realiza mediante motores paso a paso, los desplazamientos son por incrementos. El paso incremental es el mínimo desplazamiento que puede realizar la pluma. En los plotters pequeños, el paso incremental es del orden de 0,1 mm ó 0,05 mm, mientras que en los grandes puede ser de 0,025 mm ó 0,0125 mm. De esta característica depende la resolución de los dibujos.

— *Resolución:* Es una característica análoga a la anterior y se expresa también en milímetros o en pulgadas. En los electrostáticos se expresa por el número de puntos por pulgada, normalmente de 100 a 200.

— *Precisión posicional estática:* Es la



Los plotters o trazadores gráficos se utilizan profusamente en el diseño de circuitos electrónicos: para dibujar el trazado del circuito impreso, el plano de montaje de los componentes...

PLOTTERS

precisión que tiene el sistema en posicionar la pluma en unas determinadas coordenadas. Se expresa su valor absoluto en milímetros o en pulgadas.

— *Velocidad de dibujo:* Es la velocidad máxima a la que se desplaza la pluma por el papel. Se expresa en mm/seg o en pulgadas por segundo (i.p.s.). Puede ser del orden de 100 mm/seg en los plotters pequeños, y de hasta 762 mm/seg (30 i.p.s.) en los grandes. En las características se dan dos tipos de velocidades:

a) *Axial:* es la velocidad de la pluma en su desplazamiento a lo largo de su guía.

b) *Diagonal:* es la velocidad resultante en el desplazamiento combinado de la pluma y del carro o del tambor.

La velocidad total de un dibujo no sólo depende de esta velocidad máxima, sino también de otros dos factores:

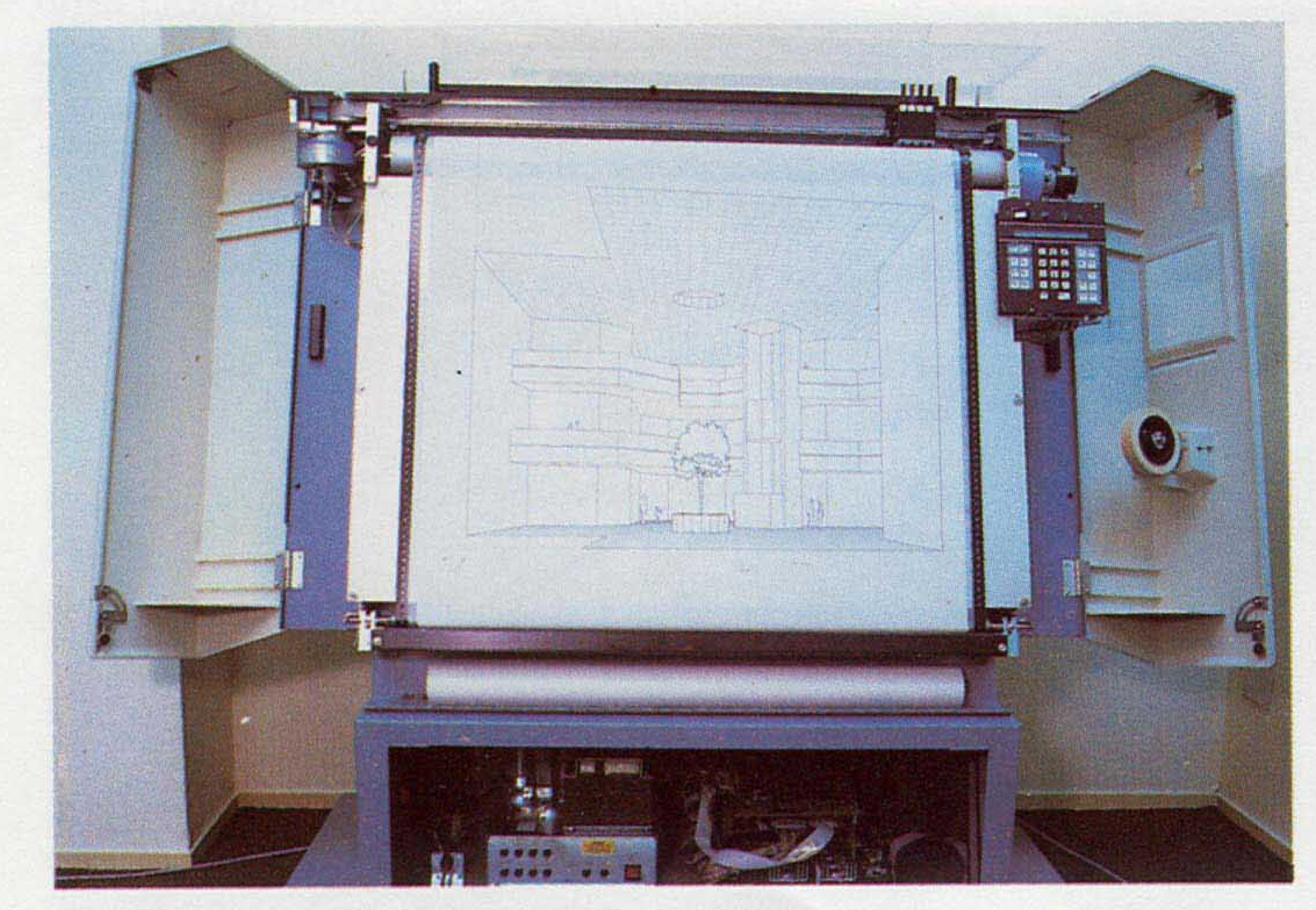
1. Aceleración: cuanto mayor sea la aceleración, antes se alcanza la velocidad máxima. Con una aceleración de 4 g se alcanza esa velocidad en una fracción de pulgada, y ello permite realizar prácticamente todo el dibujo a la velocidad máxima.

2. Tiempo de respuesta de las plumas: las plumas se aplican contra el papel mediante electroimanes y, lógicamente, tardan un tiempo tanto en subir como en bajar. Tiempos típicos de respuesta son de 2 mseg en subir y de 10 mseg en bajar.

— *Superficie de dibujo:* Son las dimensiones máximas del dibujo que puede realizarse con el plotter.

— *Número de plumas y colores:* Los plotters pueden disponer de distintas plumas de varios colores para la realización de los gráficos.

— *Funcionamiento on-line y off-line:* El plotter puede funcionar conectado directamente al ordenador (on-line), para lo cual algunos disponen de un buffer del mismo tipo que las impresoras. Sin embargo, debido a la poca velocidad de dibujo comparado con la velocidad de trabajo del ordenador, el



Plotter de tipo «flatbed» de Calcomp. El equipo de la figura es, el modelo 965, un trazador gráfico de alta velocidad que incorpora cuatro plumas de dibujo.

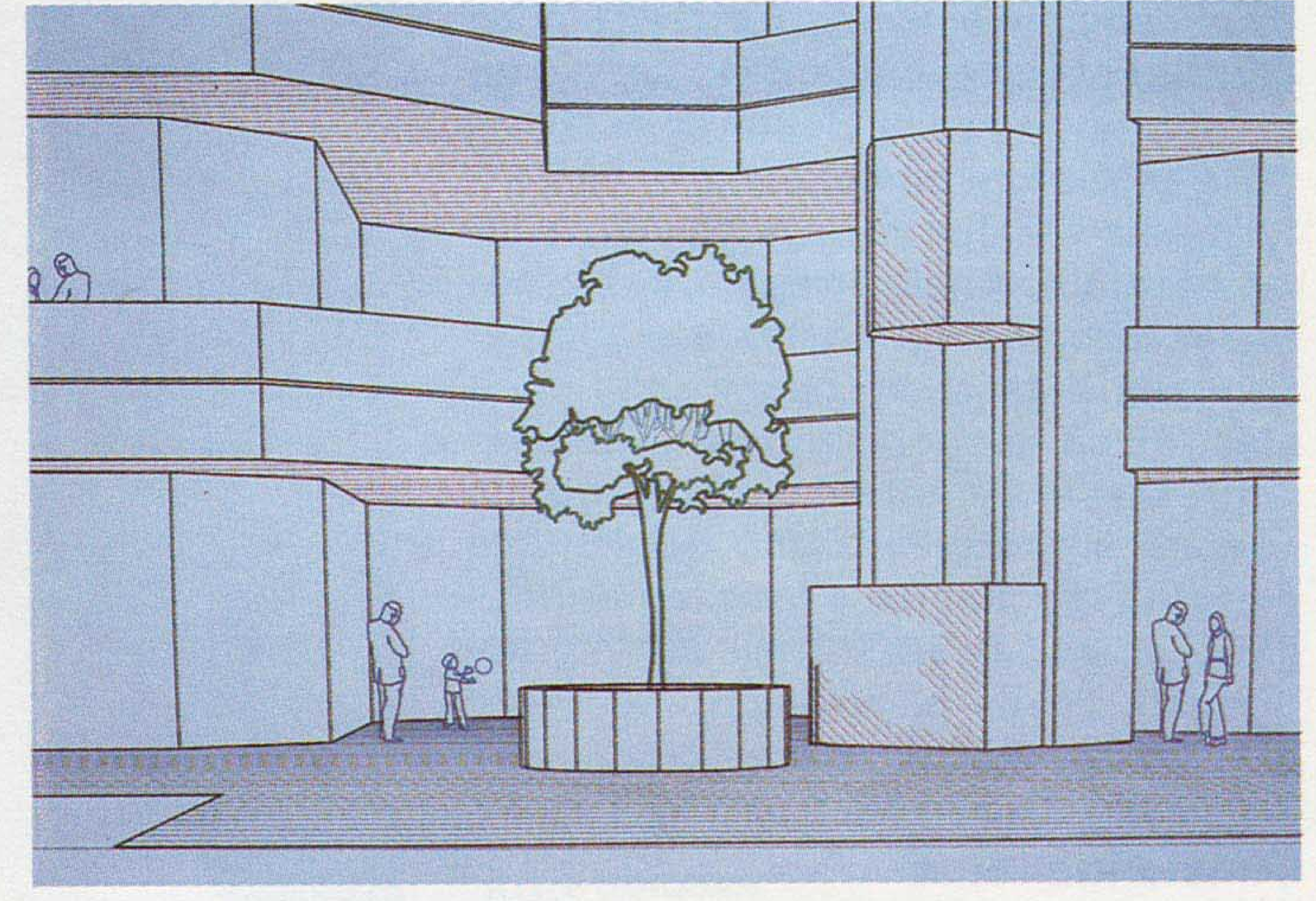


Gráfico a cuatro colores realizado con el plotter Calcomp modelo 965. La alta calidad y precisión que logran algunos de los plotters existentes en el mercado han difundido su empleo en aplicaciones de diseño técnico.



Plotter de mesa de tipo «beltbed». El modelo de la figura es el M84 de Calcomp, un trazador gráfico profesional orientado a la confección de gráficos y planos a color.



El campo de aplicación de los trazadores gráficos se incrementa día a día. De su empleo casi exclusivo en tareas de diseño técnico especializado, han saltado a aplicaciones de gestión y representación gráfica general.

funcionamiento normal de los plotters es off-line: la información correspondiente al dibujo a realizar se graba en una cinta magnética o en un disco y, posteriormente, mediante un controlador, se transfiere esa información al plotter.

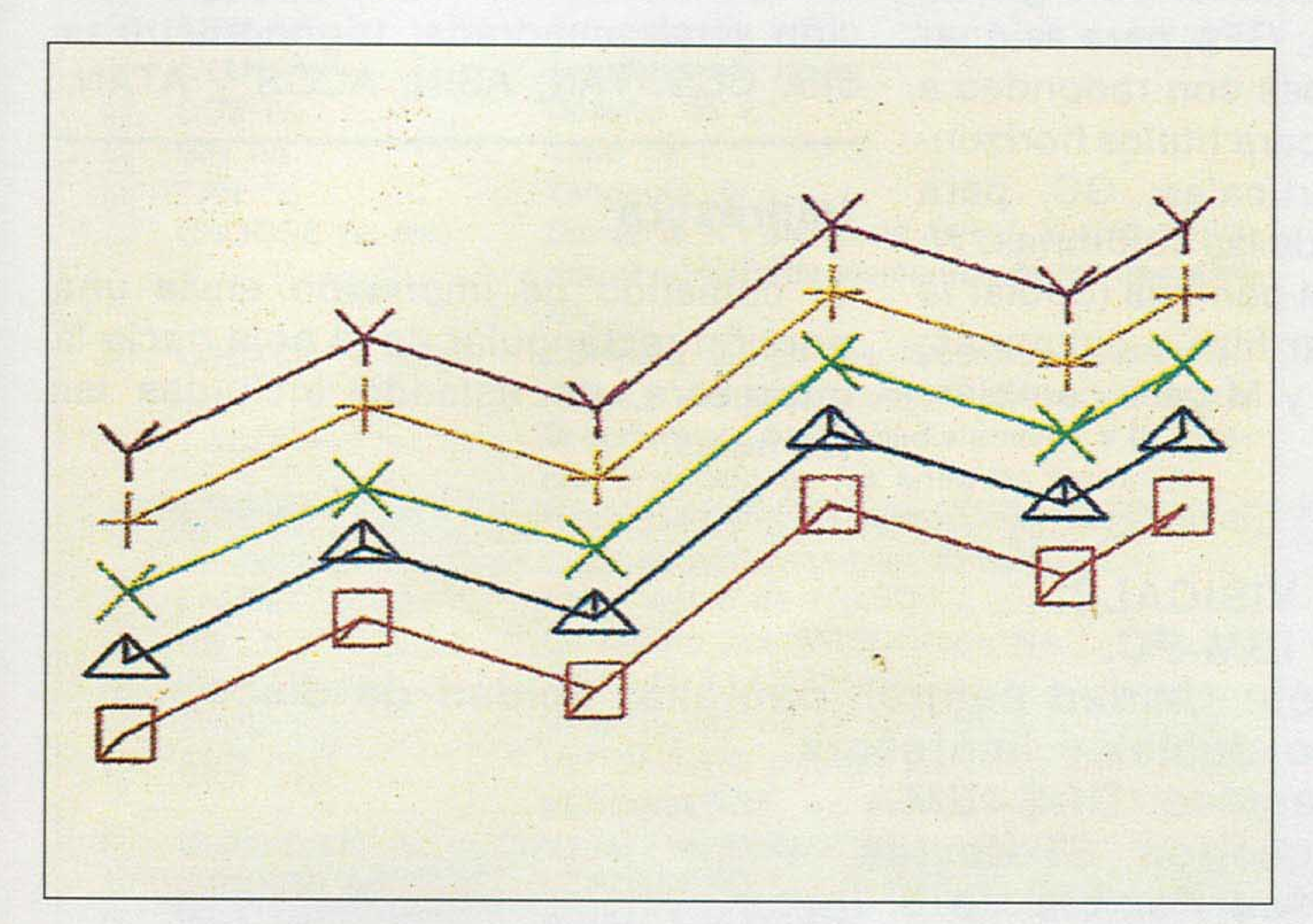
— *Programas internos:* Los plotters provistos de microprocesadores internos son capaces de almacenar programas para el dibujo de caracteres o curvas clásicas. Mediante estos programas se pueden obtener sencillamente:

1. Generación de vectores: especificando las coordenadas de un punto de destino la pluma puede ir hasta ese punto. Las coordenadas pueden ser absolutas o relativas a la posición inicial de la pluma.
2. Generación de caracteres: el programa interno es capaz de generar y dibujar caracteres a partir del código ASCII correspondiente.
3. Generación de ejes y cuadrículas: se pueden dibujar líneas continuas, de trazos, marcas, etc.
4. Sombreados y entramados: útiles para la creación de gráficas.
5. Generación de círculos y arcos: los arcos se pueden dibujar especificando el radio y los ángulos de comienzo y final.
6. Distintos tipos de líneas: las líneas se pueden dibujar contínuas, de trazos, de puntos, etc.
7. Generación de símbolos de dibujo.

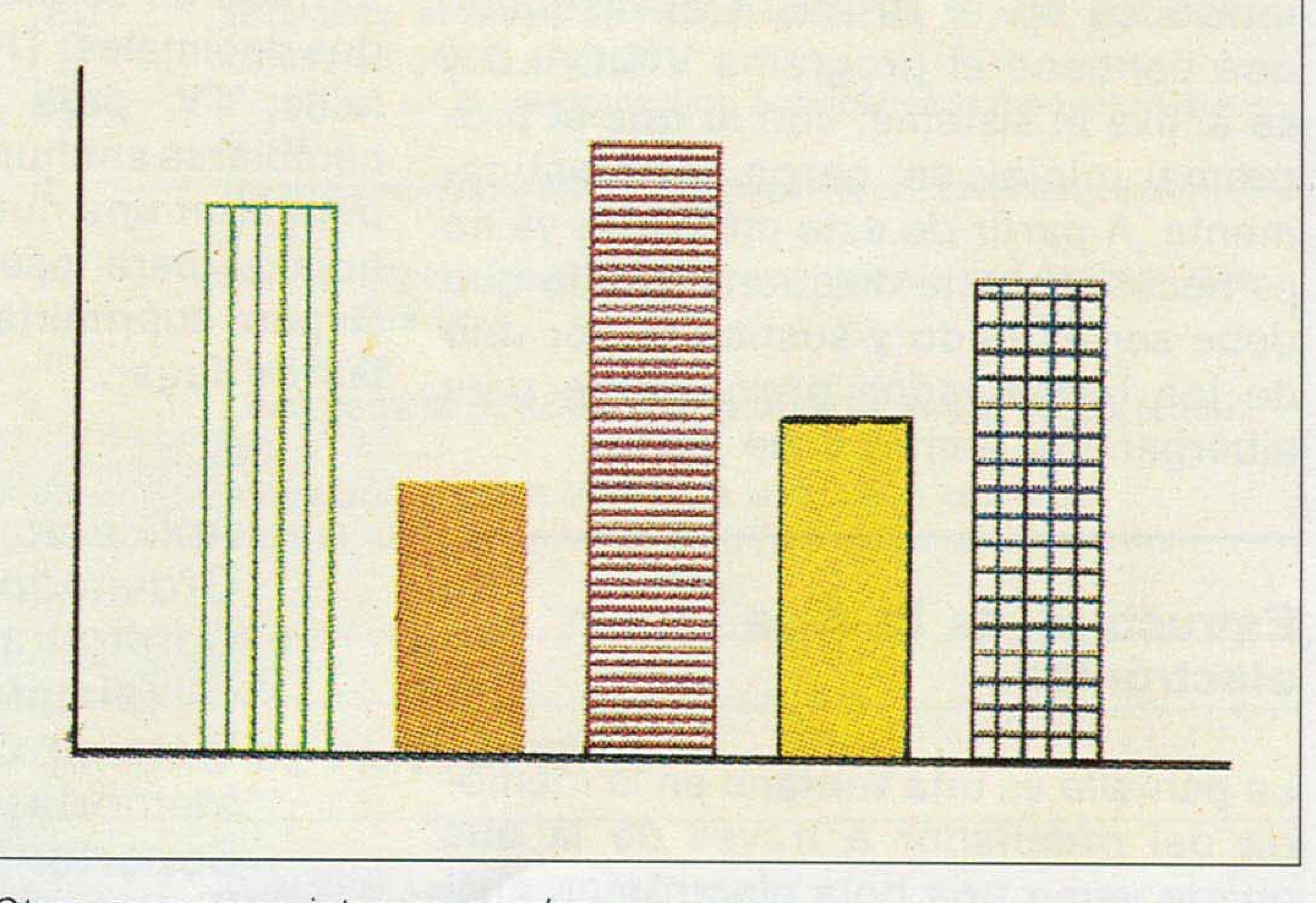
— *Tipo de interface:* Las interfaces más empleadas normalmente son:
Paralelo: puede ser del tipo centronics, como en las impresoras, o de otros tipos, por ejemplo:

RS 232.
Bucle de 20 mA.
IEEE 488.

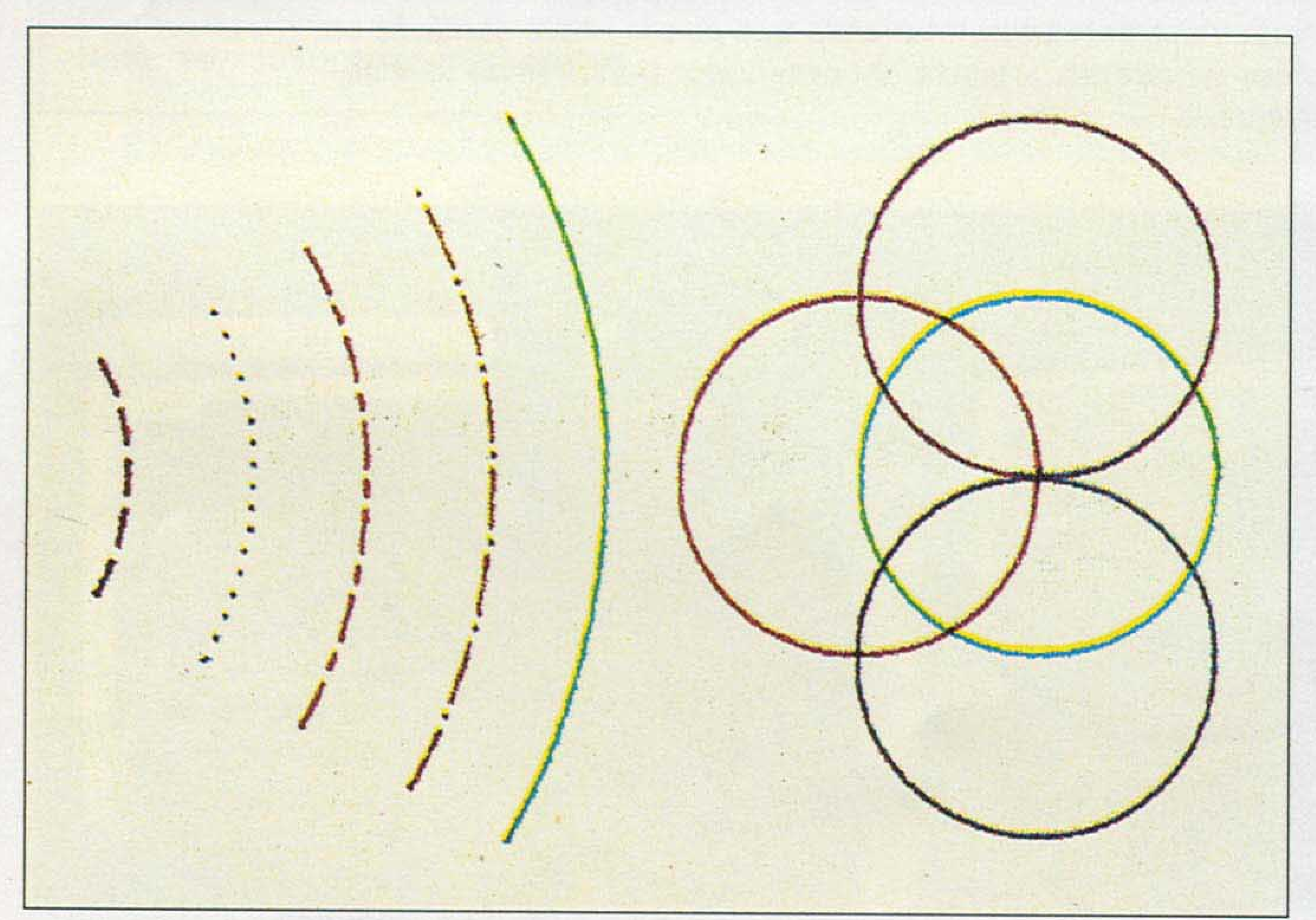
Otras características adicionales son: la tensión de alimentación (normalmente alterna), el consumo, la disipación de calor, temperatura, humedad de funcionamiento, etc.



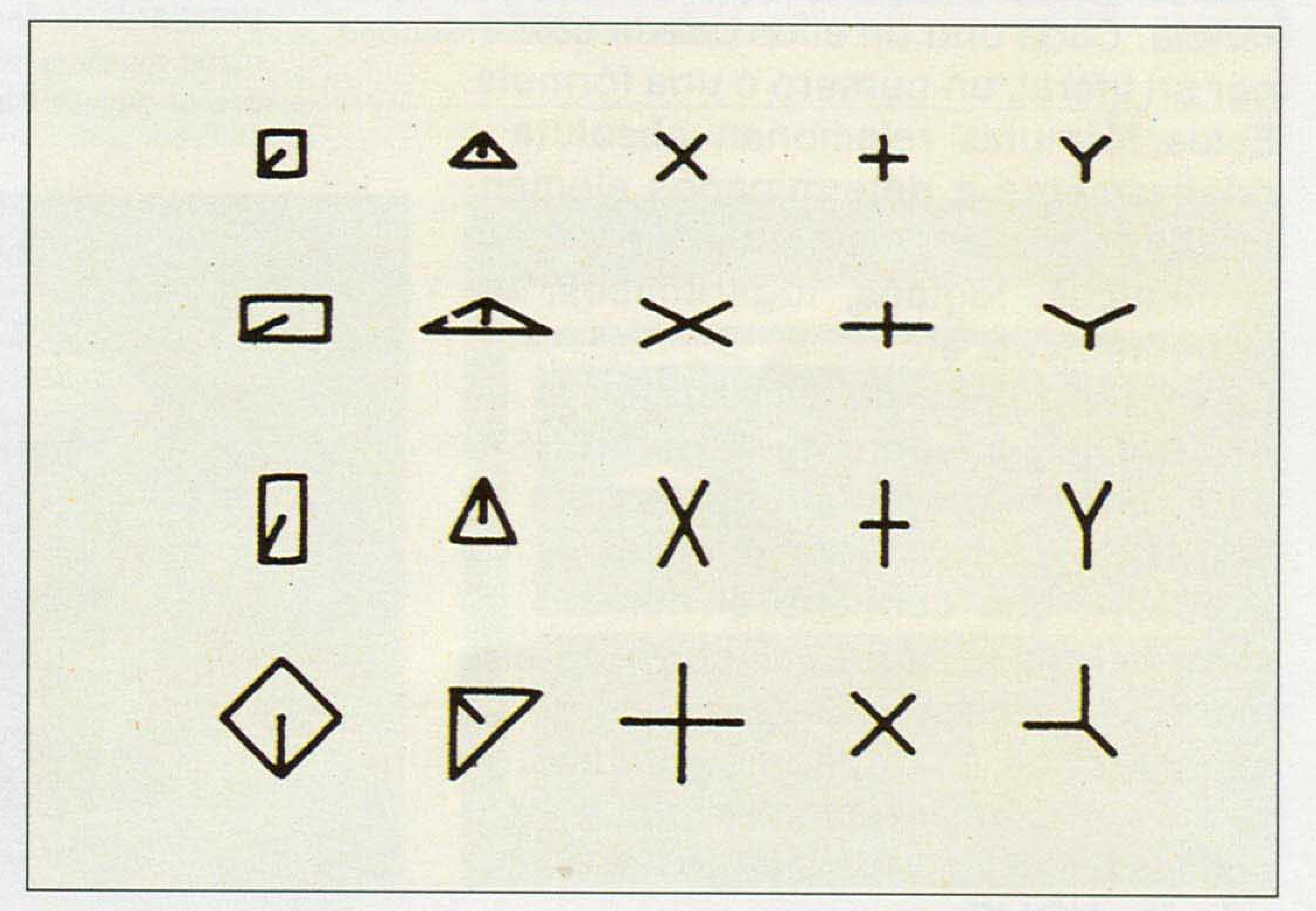
Los plotters provistos de microprocesador son capaces de almacenar programas para el trazado de curvas clásicas. Por ejemplo, para la generación de vectores.



Otros programas internos pueden ocuparse de la generación de sombreados y tramas para la confección de representaciones gráficas.



Uno de los programas internos almacenado en el plotter puede ocuparse del trazado automático de círculos y arcos de circunferencia, sin más que especificar el radio y los ángulos inicial y final.



Dentro de los programas internos, activables por comando, que suelen estar almacenados en los plotters comerciales, cabe destacar los destinados a la generación automática de símbolos y caracteres gráficos.

VISICALC

LAS hojas electrónicas surgieron al observar que muchos de los problemas de cálculo cotidianos se resolvían a base de tres herramientas casi universales: la calculadora, el lapicero y la hoja de papel. La proyección de ventas, el cálculo de impuestos, nóminas, presupuestos y balances, son algunos de los trabajos que el paquete VISICALC puede facilitar enormemente.

Comienzo de la aplicación

Para lanzar la aplicación es necesario proveerse previamente de disquettes formateados con el DOS (Sistema Operativo de Disquettes). Seguidamente, se introduce en el DRIVE A el disquette que contiene el programa VISICALC y se activa el sistema, con lo que el programa inicial se carga automáticamente. A partir de este momento ya no es necesario este disquette, por lo que debe ser extraído y sustituido por uno de los formateados previamente para albergar los archivos de datos.

Estructura de la hoja electrónica

La pantalla es una ventana en la memoria del ordenador a través de la que puede verse una hoja electrónica dividida en filas (1, 2, 3, ... 254) y columnas (A, B, C, ... BK). Cada elemento de esta matriz queda calificado por esta referencia. Cada uno de ellos puede contener un literal, un número o una fórmula. Estas fórmulas relacionan absoluta o relativamente a determinados elementos de la matriz, mediante operadores aritméticos, lógicos, logarítmicos, trigonométricos, etc.

La primera línea de pantalla es la de ENTRADA y la segunda la de MENSAJES, formando en conjunto la zona de ESTADO. La siguiente línea es la de EDICION, que contiene el número de serie del disquette. El puntero de elemento (el cursor, que inicialmente se encuentra en A1), aparece en video inverso y puede desplazarse por la hoja con las teclas de movimiento del cursor y la de HOME.

Este desplazamiento permite, lógicamente, el SCROLL, tanto horizontal como vertical de la pantalla, existiendo además la posibilidad de posicionamiento directo con el comando >, seguido de un número de fila y columna.

Teclas especiales y funciones

Son teclas especiales BKSP, para borrar caracteres, y BREAK, para detener la ejecución de un comando. Los comandos van siempre precedidos de /, como CY, para borrar la pantalla; B, para borrar toda la hoja; SS, para volcar el contenido de la memoria a disquette; R, para hacer extensiva una fórmula a varios elementos de su fila o columna de forma absoluta o relativa; GFI, para asignar a la pantalla el formato global de números enteros; GF$, para asignar formato de cantidades con redondeo a dos decimales; TH, para títulos horizontales; TV, para verticales; GC, para cambiar la anchura de las columnas; W, para abrir una nueva pantalla (doblar la hoja); I, para insertar filas o columnas; D, para suprimirlas, y M para cambiarlas de lugar.

Son funciones especiales las precedidas por @ como SUM, para calcular sumas de filas y de columnas; NA, para marcar un elemento de la matriz desconocido por el momento y con el que no se deben efectuar operaciones; MAX y MIN, para determinar máximos y mínimos, respectivamente; COUNT, para contabilizar el número de elementos ocupados en la matriz; AVERAGE, para calcular medias aritméticas; LOOKUP, para buscar determinados valores; ABS, para hallar el valor absoluto de un número, e INT, para calcular la parte entera.

Son funciones lógicas TRUE, FALSE, AND, OR, NOT e IF; logarítmicas LN y LOG10, EXP y SQRT, para exponenciación y raíz cuadrada; trigonométricas SIN, COS, TAN, ASIN, ACOS y ATAN.

Impresora

El comando de impresión envía una porción rectangular de la hoja hacia la impresora, no estando incluidas las fórmulas.

Aplicación: **VISICALC.**
Ordenador: **IBM-PC.**
Configuración: **Unidad central, pantalla, unidad de disco, (simple o doble) e impresora.**
Sistema operativo: **DOS-IBM.**
Memoria requerida: **64 Kbytes.**
Soporte: **Disco flexible de 5 1/4″.**
Documentación: **Manual de 157 páginas, en inglés.**
Copyright: **Personal Software Inc. y Software Arts Inc.**
Distribuidor: **International Business Machines, S. A. E.**

VISICALC es una hoja electrónica estandarizada que puede implementarse en otros muchos microordenadores, además del ordenador personal de IBM sobre el que se ha evaluado.



VISICALC es un programa para la generación y mantenimiento de hojas electrónicas orientadas a la resolución automatizada de cálculos tradicionalmente encomendados al lápiz y a la calculadora.

RELACION DE FUNCIONES DEL VISICALC

NOMBRES	FUNCIONES
@ ABS (v)	Valor absoluto de v.
@ AVERAGE (list)	Promedio de elementos utilizados en list, máximo 255.
@ COUNT (list)	Número de elementos utilizados en list, máximo 255.
@ EXP (v)	Exponente natural de v.
@ INT (v)	Parte entera de v.
@ LN (v)	Logaritmo natural de v.
@ LOG10 (v)	Logaritmo en base 10 de v.
@ MAX (list)	Máximo valor en list.
@ MIN (list)	Mínimo valor en list.
@ NPV (dr,range)	Valor neto actual del cash flow en range, descontado en el porcentaje especificado en la expresión dr.
@ SQRT (v)	Raíz cuadrada de v.
@ SUM (list)	Suma de valores en list.
@ ACOS (v)	Arcocoseno de v.
@ ASIN (v)	Arcoseno de v.
@ ATAN (v)	Arcotangente de v.
@ COS (v)	Coseno de v.
@ SIN (v)	Seno de v.
@ TAN (v)	Tangente de v.
@ CHOOSE (v, list)	Devuelve el v elemento de list. Si v es mayor que el número de elementos de list devuelve NA.
@ LOOKUP (v, range)	Compara v con los sucesivos valores en range y devuelve el valor correspondiente a la columna o fila, inmediatamente a la derecha o debajo de las entradas de range.
@ ERROR	Hace que todas las expresiones referentes al valor se conviertan en ERROR.
@ FALSE	Valor lógico de FALSO.
@ NA	Lo mismo que ERROR con NA.
@ PI	Valor de 3.1415926536.
@ TRUE	Valor lógico de VERDADERO.
@ AND (list)	VERDADERO si todos los valores de list son VERDADEROS; en caso contrario, FALSO.
@ IF (l, v1, v2)	V1, si l es VERDADERO; v2, si l es FALSO.
@ ISERROR (v)	VERDADERO, si v es ERROR; si no, FALSO.
@ ISNA (v)	Similar a ISERROR.
@ NOT (l)	VERDADERO, si l es FALSO; FALSO, si l es VERDADERO.
@ OR (list)	VERDADERO, si cualquier valor de list es VERDADERO; si no, FALSO.

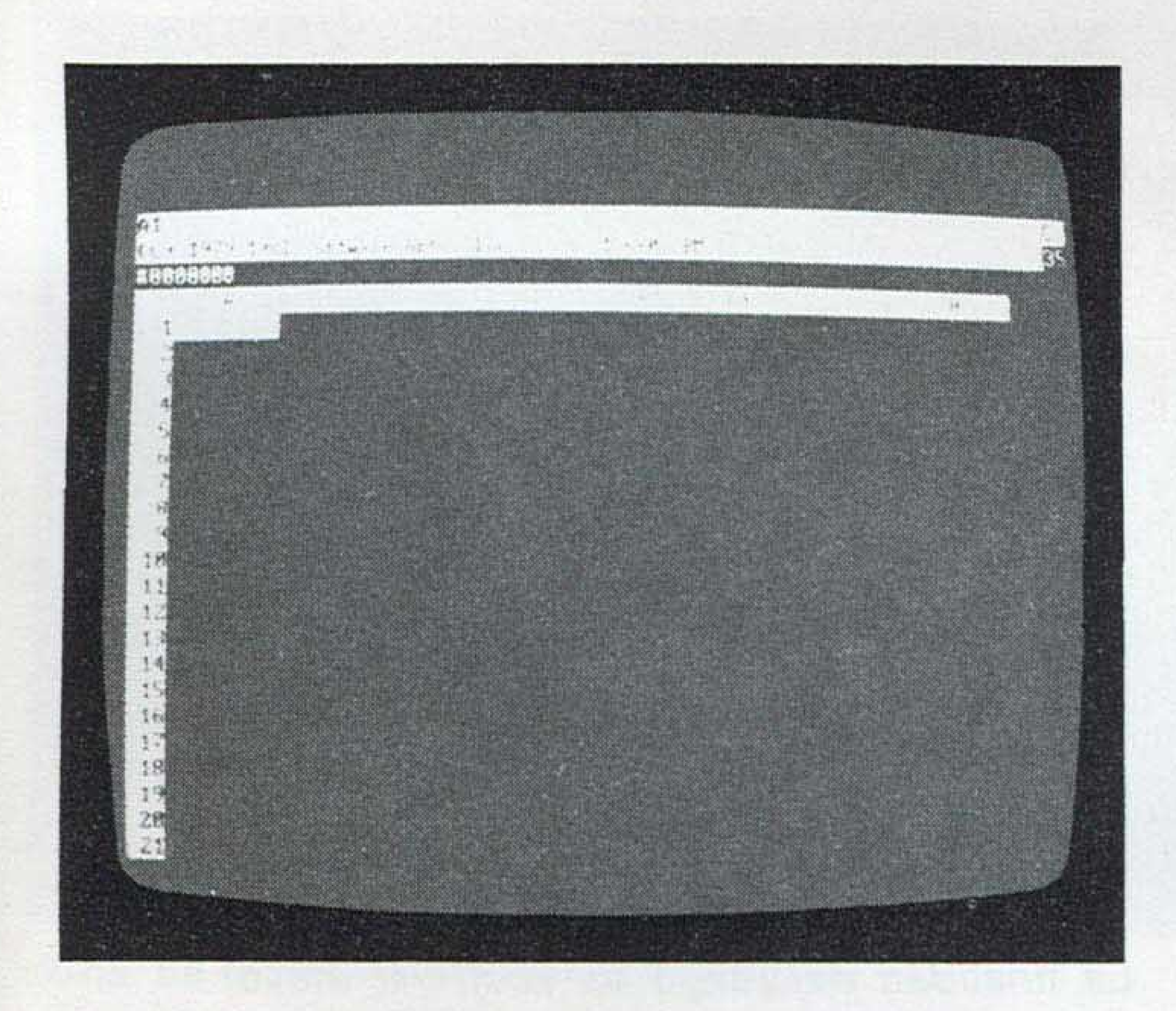
Nota: ver cuadro de parámetros.

COMANDOS DE LA APLICACION VISICALC

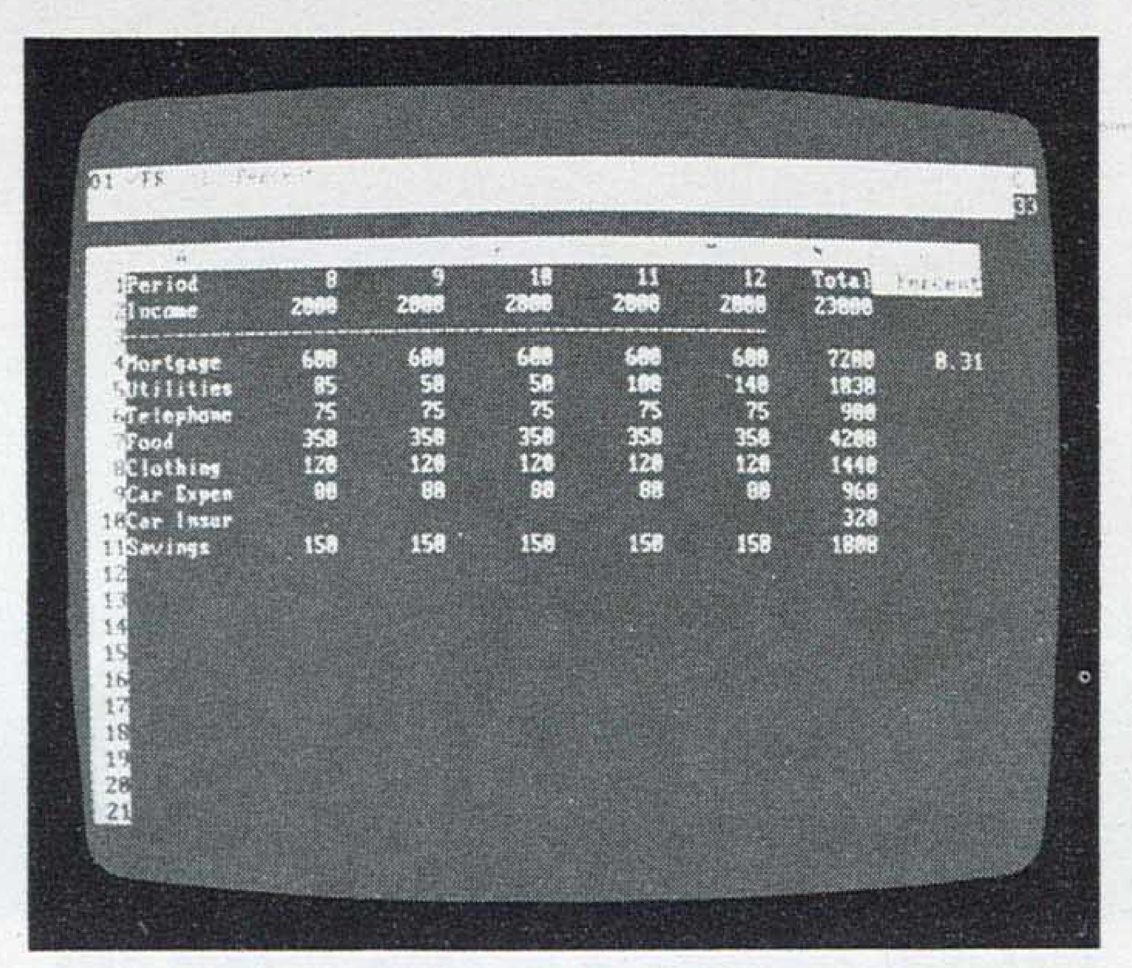
NOMBRES	COMANDOS
/B	Borra el elemento indicado por el cursor.
/C	Borra toda la hoja electrónica y devuelve pantalla simple.
/D	Borra todos los elementos en la fila o columna (/DR o /DC), en que se encuentra el cursor.
/E	Permite la completa edición del elemento señalado por el cursor.
/F	Asigna formato a la posición marcada por el cursor.
/FD	Vuelve al formato global.
/FG	Global de máxima precisión.
/FI	Formato entero
/FL	Formato alineado por la izquierda.
/FR	Formato alineado por la derecha.
/F$	Formato con redondeo a dos decimales.
/F★	Formato que reemplaza un valor por tantos asteriscos como su valor entero (útil para gráficos).
/G	Asigna formato a la totalidad de la hoja.
/GC	Asigna nuevo ancho de columna.
/GF	Asigna formato a todas las posiciones que no lo tienen específico.
/GO	Especifica el orden de recalculación de la hoja, por filas o columnas (/GOR o /GOC).
/GR	Prioridad en la recalculación, automática o manual (/GRA o GRM).
/I	Inserta una nueva fila o columna en la posición del cursor (/IR o /IC).
/M	Mueve la totalidad de una fila o columna a una posición diferente en la hoja.
/P	Impresión.
/–	Llena toda la posición con el carácter especificado (etiquetas).
/R	Afectación de una fórmula a una fila o columna.
/S	Para el almacenamiento de información en disquette.

CUADRO DE PARAMETROS

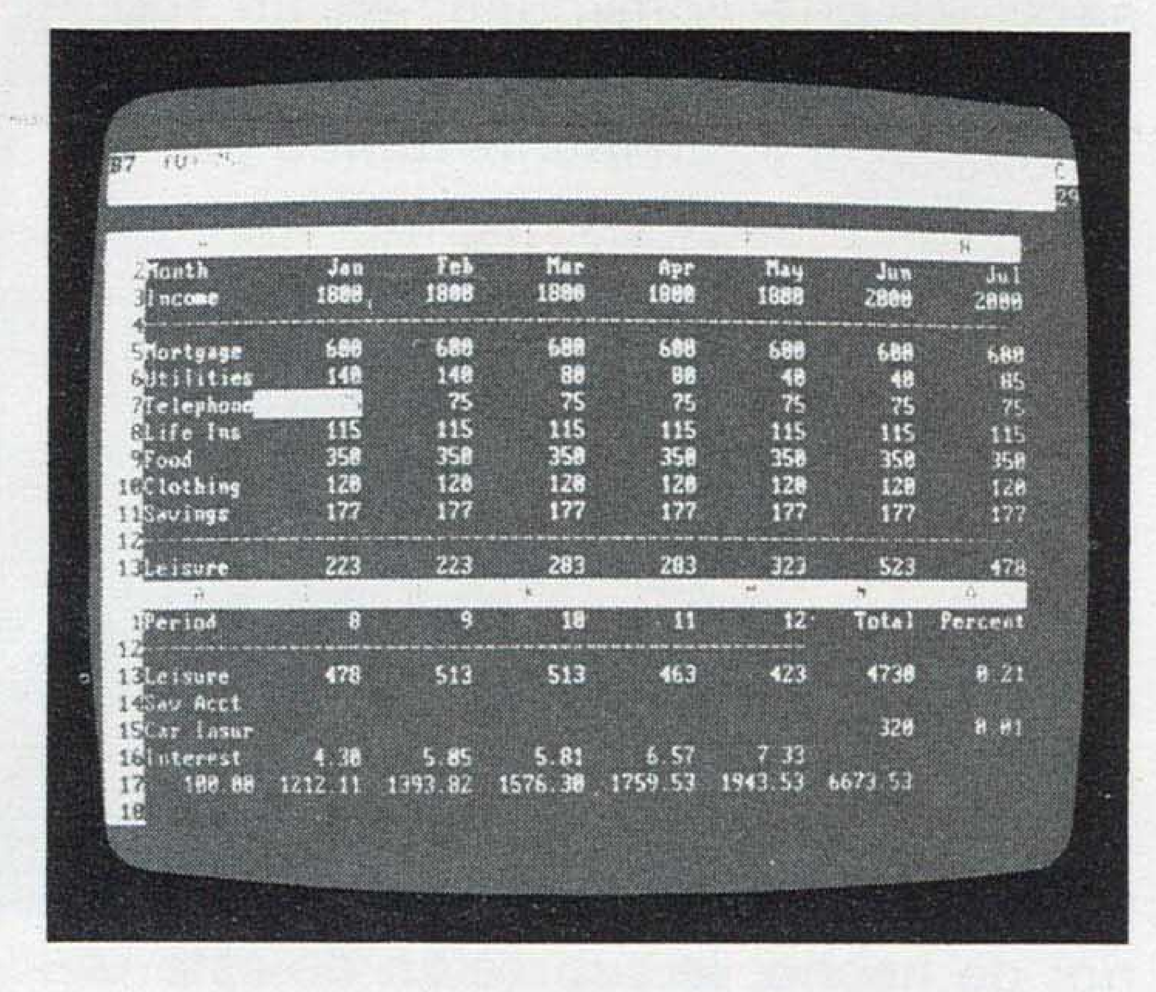
v	Cualquier valor VISICALC válido.
l	Cualquier valor lógico.
list	Cualquier combinación de valores y series separadas por comas.
range	Porción de fila o columna especificada por la coordenada inicial, período y coordenada final.



La confección de presupuestos, el estudio de la evolución de ventas, el cálculo de nóminas... son tareas que la aplicación VISICALC facilita y automatiza.



La pantalla corresponde a una ventana abierta en la memoria del ordenador a través de la que se visualiza la hoja electrónica cuyos campos están coordenados en filas y columnas.



Las posibilidades de cálculo de las hojas electrónicas permiten operar su contenido. En la pantalla aparece una hoja electrónica para el control de la economía doméstica.

PROGRAMA

Nombre: ***Antiaéreo***
Ordenador: ***Sinclair ZX-81***
Memoria necesaria: ***16 Kbytes***
Lenguaje: ***BASIC***

Este juego, programado inicialmente para el ZX-81, presenta una total compatibilidad con el modelo ZX-Spectrum; en este último puede incrementarse la vistosidad del juego, dadas sus mayores posibilidades de color y sonido, así como de caracteres gráficos.

El objetivo del juego es abatir el mayor número de aviones posible con los cincuenta proyectiles de que se dispone. Hay que tener en cuenta que, como es habitual, hasta que los misiles no alcancen el tope superior, no se puede volver a disparar. La puntuación que se obtiene en cada acierto está en relación a la dificultad de hacer blanco. Así, por ejemplo, tiene más valor abatir un avión cuanto más arriba de la pantalla haya aparecido; análogamente, proporciona mayor puntuación utilizar los proyectiles de los bunkers 1 y 2 que los del 3 y, a su vez, los del 1 más que los del 2. Para el lanzamiento de proyectiles, se presionan las teclas 1, 2 y 3, atendiendo a la batería desde la que se desea disparar.

Al final del juego, se compara el número de puntos obtenidos con el récord que se tenga establecido hasta el momento, informándose oportunamente de si éste ha sido batido o no.

No es éste el caso de uno de esos programas que brillan por la genialidad de alguna de sus instrucciones, sino que es más bien uno de esos ejemplos de sencillez que consiguen, en un reducido espacio de memoria, realizar un juego lo suficientemente rápido y atractivo. Es por esto por lo que se puede considerar como un programa ideal, para ser traducido a código máquina por aquellos que comiencen en esa, a veces difícil, tarea.

Si se suprimen las instrucciones 300 y 310, de una aparente simplicidad, se observará la gran importancia de las mismas. En este caso, el bucle en vacío tiene la misión de igualar velocidades, ya se encuentre un proyectil en curso o no; de hecho, la supresión de este retardador hace muy difícil el impacto, anulando prácticamente la operatividad del juego.

Estructura del programa

Líneas	Comentario
10-90	Inicializaciones.
100-150	Comienzo de bucle principal y toma de datos.
160-200	Lanzamiento del proyectil.
210-240	Movimiento del proyectil.
250-290	Averigua si el proyectil alcanzó el tope superior.
300-340	Retardador y fin de bucle principal.
350-420	Avión alcanzado.
430-510	Fin del programa.

Cuadro de variables

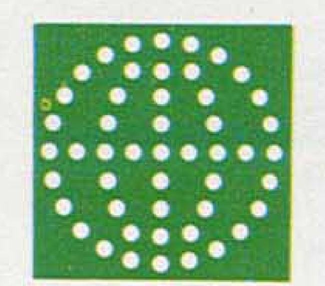
Variable	Descripción
A	Fila de impresión del avión.
B	Columna de impresión del proyectil.
D	Switch de proyectil en curso.
H	Fila de impresión del proyectil.
L	Columna de impresión del avión.
M	Munición.
P	Puntos.
R	Récord.
T	Retardador FOR para igualación de velocidades.

```
  10 REM LOPEZ MARTINEZ
  20 LET R=0
  30 CLS
  40 PRINT AT 21,0;"████████████████████████████████"
  50 PRINT AT 20,5;"▄█▄";TAB 15;"▄█▄";TAB 25;"▄█▄"
  60 PRINT AT 0,0;"PUNTOS: 0","MUNICION: 50"
  70 LET D=0
  80 LET P=0
  90 LET M=50
 100 LET A=INT (RND*14)+2
 110 FOR L=0 TO 28
 120 PRINT AT A,L;" ▙▬"
 130 IF D THEN GO TO 210
 140 IF INKEY$="" THEN GO TO 300
 150 IF INKEY$<"1" OR INKEY$>"3" THEN GO TO 300
 160 LET B=10*VAL INKEY$-4
 170 LET D=1
 180 LET M=M-1
 190 LET H=19
 200 PRINT AT 0,26;"  ";AT 0,25;M
 210 LET H=H-1
 220 IF H<>A THEN GO TO 240
 230 IF B>L AND B<L+4 THEN GO TO 350
 240 PRINT AT H,B;"!";AT H+1,B;" "
 250 IF H>2 THEN GO TO 320
 260 LET D=0
 270 PRINT AT H,B;" "
 280 IF NOT M THEN GO TO 430
 290 GO TO 320
 300 FOR T=0 TO 1
 310 NEXT T
 320 NEXT L
 330 PRINT AT A,29;"   "
 340 GO TO 100
 350 PRINT AT H+1,B;" "
 360 PRINT AT A,L;"BOUM"
 370 PAUSE 1000
 380 LET D=0
 390 PRINT AT A,L;"    "
 400 LET P=P+(16-A)*(4-((B-6)/10+1))
 410 PRINT AT 0,8;P
 420 IF M THEN GO TO 100
 430 IF P<=R THEN GO TO 470
 440 PRINT AT 9,5;"ESTABLECIO NUEVO RECORD"
 450 LET R=P
 460 GO TO 480
 470 PRINT AT 9,9;"EL RECORD SIGUE"
 480 PRINT AT 11,16-(10+LEN STR$ R)/2;"EN ";R;" PUNTOS"
 490 PRINT AT 13,3;"DESEA INTENTARLO DE NUEVO ?"
 500 IF INKEY$="S" THEN GO TO 30
 510 IF INKEY$<>"N" THEN GO TO 500
```

Aunque el programa se ha confeccionado para el ZX-81, resulta totalmente compatible con el ZX-Spectrum. La ejecución en este último microordenador incrementará la espectacularidad del juego al poder utilizar color y sonido.



La finalidad del juego es abatir el mayor número posible de aviones enemigos con los cincuenta proyectiles disponibles en el antiaéreo. La puntuación dependerá de la dificultad de cada uno de los blancos realizados.



EL TELETRABAJO

CUÁNTAS veces hemos soñado, mientras al ir al trabajo nos encontramos en pleno atasco, en poder trabajar en nuestra propia casa? ¡Si pudiéramos evitar tantas horas de transporte y no tener que levantarnos dos o tres horas antes del comienzo de la jornada laboral...!

¿Qué es el teletrabajo?

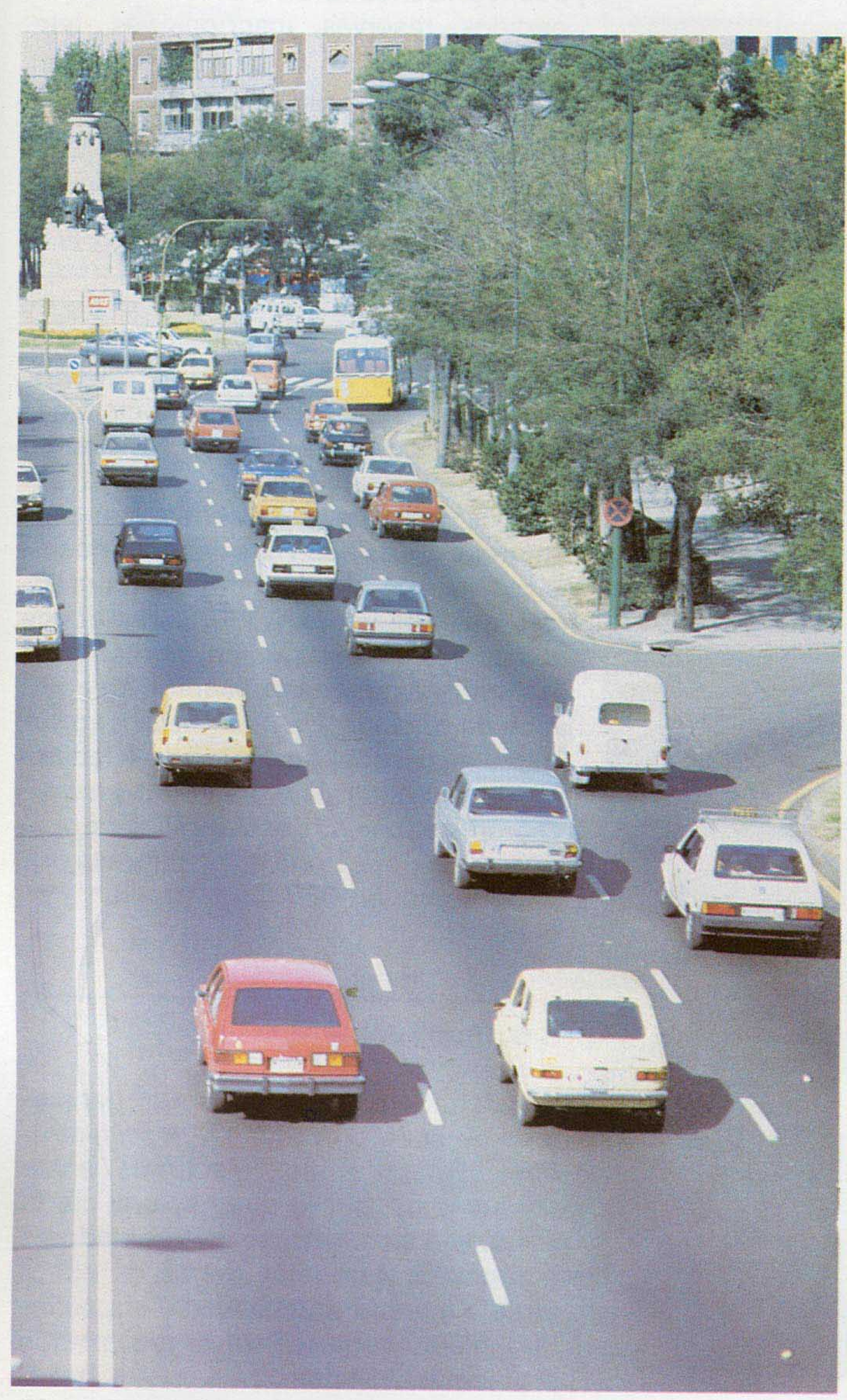
Se puede definir como «el trabajo realizado por una unidad (persona o grupo) descentralizada, es decir, separada de su oficina normal y cuya actividad implica el uso de medios de comunicación».

El teletrabajo es posible, sobre todo en el sector terciario, gracias a la conjunción de medios informáticos y telemáticos. La primera experiencia notable del teletrabajo se inició en Gran Bretaña, en 1962, en una empresa de consultores de informática. Actualmente, la citada empresa cuenta con unos 600 empleados que trabajan en sus domicilios utilizando terminales. Otras experiencias se han realizado, posteriormente, en Estados Unidos y en Francia, dentro de los sectores bancarios, informáticos, telefónicos y de secretariado.

Los resultados de las experiencias son satisfactorios. Por ejemplo, se anuncia ya que todos los empleados americanos de la American Telephon and Telegraph trabajarán en sus domicilios en 1990. En Francia se prevé que, antes de acabar esta década, serán 400.000 las personas que trabajarán de esta forma.

Los medios de teletrabajo

Los medios que necesita el teletrabajo son muy variados:



Una de las ventajas del teletrabajo se hace evidente cuando nos encontramos en pleno atasco en el desplazamiento al puesto de trabajo. No cabe duda de que el ahorro en tiempo de transporte resulta un factor a considerar.



La evolución de los medios informáticos y telemáticos hacen del teletrabajo una alternativa no sólo posible sino real en nuestros días. Son ya muchas las experiencias en curso tanto en Europa como en los EE. UU.



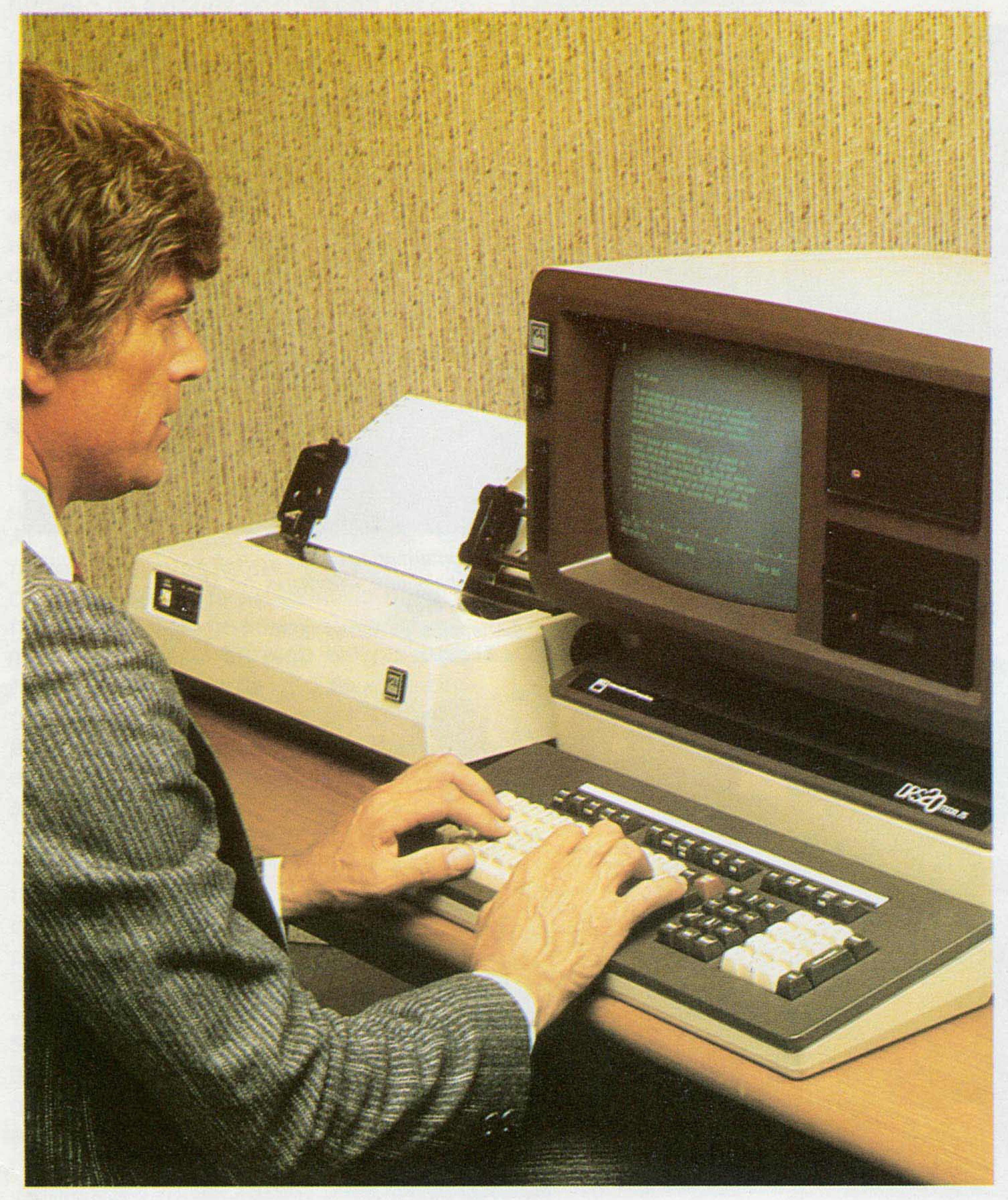
Los medios puestos en juego para el teletrabajo pueden servir incluso para acercar al propio domicilio del administrado los servicios de la Administración, con un indudable aumento en la eficacia y rapidez en la gestión de los servicios; las largas y tediosas colas de espera pueden desaparecer de las oficinas públicas.

EL TELETRABAJO

— Redes de transmisión de datos.
— Teléfonos y sus accesorios (contestador automático, registradores, etc.).
— Télex.
— Equipos de tratamiento de textos.
— Videotext y teletext.
— Terminales de pantalla y otros equipos periféricos.
— Medios de telepago.
— Videoteléfono.
— Telecopiadoras.

Estos medios deben de ubicarse en locales que actualmente son de tres tipos:

El domicilio del empleado. El equipo es proporcionado por la empresa; o bien el trabajador es autónomo e instala su propio equipo conectándose con las empresas con las cuales contrata.
La teleagencia. Un local abierto al público con empleados que sirven de intermediarios entre los clientes y la organización. Pueden tener mucho futuro como agencias para la administración pública.
El telelocal. Locales de una o varias empresas equipados con el material conveniente para acercar sus servicios a los clientes.



La implantación del teletrabajo pasa por la reorganización de la empresa. En cualquier caso, pueden ya vislumbrarse las grandes posibilidades que tiene de convertirse en una de las formas de trabajo más importantes de la nueva sociedad tecnológica.

Tareas posibles

En el momento actual los trabajos que pueden deslocalizarse son:

— *Tareas de tratamiento de textos.* Composición, corrección, difusión, etc.

— *Tareas de gestión administrativa.* Consulta y mantenimiento de archivos y otras tareas que actualmente se realizan a través de terminales.

— *Tareas que implican relación con el público.* Consulta de bancos de datos, pedidos, reservas, inscripciones, telepagos, etc.

— *Tareas informáticas.* Especialmente programación.

Ventajas...

El teletrabajo ahorra tiempo de traslado, que puede dedicarse al ocio o a la familia y, simultáneamente, revierte en un ahorro energético. Permite el acceso al trabajo a las personas, como las mujeres con niños y los disminuidos físicos, que de otra forma tendrían grandes dificultades para trabajar. Reduce los costos de producción, al disminuir las inmovilizaciones de capital en edificios de oficinas. Da una nueva oportunidad a personal desmotivado y facilita la reconversión de empleo por amortización de tareas. Descentraliza la gestión al máximo, tanto en la empresa privada como en la Administración Pública.

...e Inconvenientes

El teletrabajo implica una reorganización de la empresa y de la Administración. Dificulta las relaciones entre compañeros, por lo que hay una cierta tendencia a convertirse en trabajadores autónomos. Los jefes deben de adaptarse a mandar a distancia. Pero es indudable que una vez solventados estos problemas, el teletrabajo puede hacer compatible un trabajo eficiente con un verdadero tiempo libre para el ocio y la cultura. ¿Va a ser ésta la forma de trabajar de los años noventa?...